1974-12

Improvement of AN/TPQ-27 filter and control techniques.

Lentz, Robert Eugene

Monterey, California. Naval Postgraduate School

IMPROVEMENT OF AN/TPQ-27 FILTER AND CONTROL TECHNIQUES

Robert Eugene Lentz

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

IMPROVEMENT OF AN/TPQ-27 FILTER
AND CONTROL TECHNIQUES

by

Robert Eugene Lentz

December 1974

Thesis Advisor: H. A. Titus

Approved for public release; distribution unlimited.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Improvement of AN/TPQ-27 Filter and Control Techniques		5. TYPE OF REPORT & PERIOD COVERED Electrical Engineer December 1974
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Robert Eugene Lentz		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		12. REPORT DATE December 1974
		13. NUMBER OF PAGES 201
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) Naval Postgraduate School Monterey, California 93940		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Kalman Filter Application
AN/TPQ-27
Remote Guidance and Control
Radar Tracking

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A modified linear Kalman filter with deterministic forcing is used to improve the tracking capabilities of the Marine Corps AN/TPQ-27 remote tactical aircraft guidance system. Both sixth and ninth order filters are developed and used with the Precision and Coarse Guidance simulation programs. A technique for overcoming the effects of autopilot bias is presented and tested through simulation.

DD FORM 1473, 1 JAN 73 (Page 1) EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601 | 1

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Block #20 Continued

Precision Guidance control is modified to utilize angle error and angle error rate to generate corrective commands. Significant improvement in aircraft state estimation, bombing accuracy, and overall system response is shown. The Coarse Guidance algorithms and simulation program are nearly completely new and perform aircraft guidance to the bombing run with more than adequate precision under simulated conditions. The new version of the program is significantly less complex than the previous version and incorporates features which more realistically reflect actual conditions under which the system would be used.

Improvement of AN/TPQ-27

Filter and Control Techniques

by

Robert Eugene Lentz
Lieutenant, United States Navy
B.S.E.E., University of Maryland, 1968

Submitted in partial fulfillment of the
requirements for the degree of

ELECTRICAL ENGINEER

from the

NAVAL POSTGRADUATE SCHOOL
December 1974

ABSTRACT

A modified linear Kalman filter with deterministic forcing is used to improve the tracking capabilities of the Marine Corps AN/TPQ-27 remote tactical aircraft guidance system. Both sixth and ninth order filters are developed and used with the Precision and Coarse Guidance simulation programs. A technique for overcoming the effects of autopilot bias is presented and tested through simulation. Precision Guidance control is modified to utilize angle error and angle error rate to generate corrective commands. Significant improvement in aircraft state estimation, bombing accuracy, and overall system response is shown. The Coarse Guidance algorithms and simulation program are nearly completely new and perform aircraft guidance to the bombing run with more than adequate precision under simulated conditions. The new version of the program is significantly less complex than the previous version and incorporates features which more realistically reflect actual conditions under which the system would be used.

TABLE OF CONTENTS

I. INTRODUCTION

A. AN/TPQ-27 INTRODUCTION

The AN/TPQ-27 system is a tactical aircraft guidance and control system used by the Marine Corps to guide strike force aircraft along a preplanned route, and then down a final leg for the purpose of performing precision bombing.

The mission control and guidance is divided into two modes. The first mode, called Coarse Guidance, takes the aircraft from a TACAN entry point through a series of straight paths and command turns. These "legs" of the Coarse Guidance run are defined by the leg end points. The actual route to be followed is determined by tactical considerations. The last leg is also the bombing leg. After proper entrance onto the final approach to the bombing leg, the second mode of the mission guidance control takes command; this is called Precision Guidance. During the Precision Guidance mode, very accurate aircraft position estimates are developed to enable bomb placement calculations which will be accurate to errors on the order of feet at the final impact point. The primary difference between these two modes, other than purpose, is the radar precision used to make the measurement of aircraft position. The resultant precision is also a function of the data rates of the respective radars, which differ significantly in this case.

The initial efforts on this system have suffered from some problems which are also apparent from observation of the

results of software simulations of actual missions. The purpose of this study was to look into various techniques currently employed in the simulation programs, and attempt to improve the response and overall accuracy of the system through the use of different and/or improved algorithms.

B. AN/TPQ-27 PROBLEM AREA DEFINITION

The specific problem areas which were investigated were defined primarily through contact with personnel closely associated with the system's performance. Other areas were noted as in need of improvement during the familiarization and simulation trial phases of study.

1. Problem Areas in Coarse Guidance

Coarse guidance has suffered from many separate but related problems. Pilots have complained that the controls sent to the aircraft in the final moments before turning to a new leg have been violent. In addition, there have been complaints of not knowing where the aircraft was at any time other than having just exited from a "command turn," i.e., a turn from one leg onto a new leg, vice a course correction. The simulation program was found to be suffering from unnecessary complexity in some areas, and was apparently in need of refined estimation and control procedures.

2. Problem Areas in Precision Guidance

The single most prevalent complaint with the Precision Guidance program was the length of time required for the estimates to stabilize in order than an accurate determination on exactly when the bomb should be dropped could be made.

Again, this and other related problems with Precision Guidance seemed due to inadequate aircraft estimation algorithms, and a control scheme which was too simplistic in design.

II. AIRCRAFT POSITION AND VELOCITY ESTIMATION

A. BACKGROUND AND INTRODUCTION TO THE KALMAN FILTER

The technique previously used to provide noise filtering on aircraft position and velocity was a standard alpha-beta filter with parameters chosen to yield the "optimal" tracking capability in accordance with the theory developed in [1]. However, this reference also states that in the case when adaptive tracking is required, the α parameter should be permitted to vary with observed high frequency power fluctuations in the error signal

$$e = x_n - x_{pn} \tag{1}$$

where x_n is the state estimation, and x_{pn} is the state prediction prior to measurement. Provisions for this variation were not included in the tracking algorithm which was implemented. In addition, the alpha-beta filter which was implemented was not an unbiased estimator of the aircraft state vector. This is due to the fact that as controls were used to cause changes in the free inertial model of the motion assumed by the alpha-beta filter, no corresponding changes were added to the filter states to account for this deterministically added control. This accounts for the exceptionally large and prolonged transient errors which resulted from large control bank commands.

The Kalman filter yields a minimum variance estimate of the state vector when the statistics of the noise are as

described below. This filter includes the effects of deterministic control commands to the aircraft to yield an estimator which is very nearly unbiased. The greatest improvement in estimation is yielded during the initial filter transient behavior. This is particularly critical in this application, in order to overcome the long filter settling period required by the alpha-beta filter.

B. KALMAN FILTER ASSUMPTIONS AND GENERAL RECURSION EQUATIONS

Application of the Kalman filter assumes that the discrete system under consideration satisfies

$$X(k+1) = \phi(k)X(k) + W(k) \tag{2}$$

$$Z(k) = H(k)X(k) + V(k) \tag{3}$$

where X is an n x 1 state vector, Z is an m x 1 output vector, W is a zero-mean n x 1 vector of state excitation white noise, uncorrelated with the zero-mean additive white noise vector V, ϕ is the state transition matrix (n x n), and H is the m x n observation or measurement matrix. The assumed noise statistics are

$$E[V(k)\ V(j)^T] = R(k)\ \delta(k,j) \tag{4}$$

$$E\{\Gamma[W(k)\ W(j)^T]\Gamma^T\} = Q(k)\ \delta(k,j) \tag{5}$$

$$E[V(k)\ W(j)^T] = 0 \text{ for all } k,j \tag{6}$$

where

$$\delta(k,j) = \begin{cases} 0 & k \neq j \\ 1 & k = j. \end{cases} \tag{7}$$

The actual Kalman filter recursion equations are summarized

below, where $\hat{X}(k/j)$ denotes the estimate of the state $X(k)$ based upon the j measurement observations $Z(1), Z(2), \ldots, Z(j)$.

$$P(k/k-1) = \phi(k,k-1)P(k-1/k-1)\phi(k,k-1)^T + Q(k) \tag{8}$$

$$G(k) = P(k/k-1)H(k)[H(k)P(k/k-1)H(k)^T + R(k)]^{-1} \tag{9}$$

$$P(k/k) = P(k/k-1) - G(k)H(k)P(k/k-1) \tag{10}$$

$$\hat{X}(k/k) = \hat{X}(k/k-1) + G(k)[Z(k) - H(k)\hat{X}(k/k-1)] \tag{11}$$

$$\hat{X}(k/k-1) = \phi(k,k-1)\hat{X}(k-1/k-1) + \Gamma(k)U(k-1) \tag{12}$$

where $P(k/k-1)$ is the covariance of error for the state prediction vector $\hat{X}(k/k-1)$, $P(k/k)$ is the covariance of error matrix for the state estimation vector $\hat{X}(k/k)$, and $G(k)$ is the gain matrix to be applied at the time of the k^{th} measurement. Further detail on the summary and development of the Kalman filter equations is available in [2], [3], and [4].

C. SELECTION OF REQUIRED SYSTEM MODEL ORDER

The order of the filter refers to its capability to track a target exhibiting a particular type of motion without error, in a noiseless environment. For a single dimensional problem, a first order filter would estimate position only. Second order filters are capable of tracking a constant velocity target and estimating both position and velocity. Similarly, third order filters are capable of tracking a target exhibiting a constant acceleration profile, estimating position, velocity and acceleration in the process.

Documentation provided on the original AN/TPQ-27 programs indicated that the aircraft would be flying a constant airspeed

profile. This then indicates that the aircraft model might be appropriately chosen to be a free inertial ($1/s^2$) plant. Command orders to the aircraft in the form of bank angles serve to change the aircraft's heading in a totally deterministic manner, provided the true transfer function of the aircraft is known with respect to roll response. In view of the above, it was originally thought that it would be sufficient to implement second order Kalman filters to estimate position and velocity in each of the three coordinates, yielding a sixth order filter.

It was later discovered that a problem known as autopilot bank angles bias exists with sufficient frequency and resultant imprecision that the order of the original feedback loop was increased to compensate for the error through the use of a discrete integrator [5]. Concepts such as integrators and digitally implemented lead-lag networks were to be avoided in the improved version of the simulation routines due to the resultant phase lags which they introduce. This is not to say that the steady-state response is in error, but simply that the time to reach that state is unsatisfactorily long. When simulations were run with non-zero autopilot bias bank angles in conjunction with second order Kalman filters, significant errors resulted.

A bias angle causes the aircraft to turn in a given direction at a constant heading rate. It was postulated that it might be possible to estimate the bias angle, and induce an anti-bias in the estimator, but the noise on the bias estimate proved to be excessive.

Third order filters will estimate a constant acceleration. The second attempt to overcome the bias problem was to postulate that the acceleration in the horizontal components would not change significantly over the relatively short periods of flight time in question. (Of course, if a bias existed and the aircraft was flown for a long enough time, the path flown would appear as a circle, and the third order filters could not possibly be satisfactory for that situation.) Thus, development of two separate filtering schemes was pursued. A sixth order (second order for each of three dimensions of motion) and a ninth order (third order for each of the three dimensions) filter were developed and tested. Use of the programs is very similar for each of the filters and is described below.

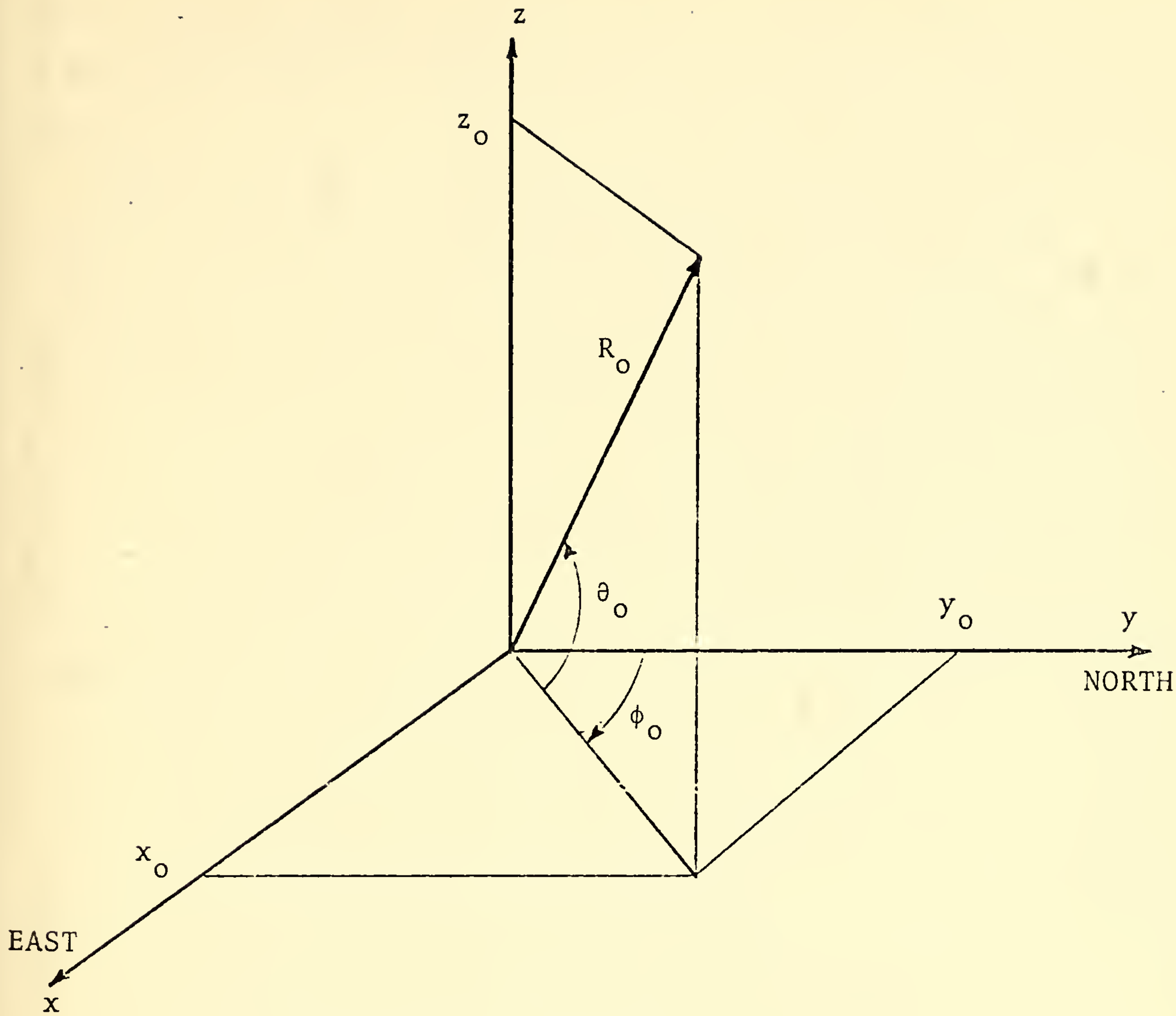
D. DERIVATION OF THE COVARIANCE OF MEASUREMENT NOISE MATRIX, R(k)

The Kalman Filter assumptions include linear relationships among measurements and states, as well as linear state transition dynamics. The first of these is of concern at this point. The states of concern in measurement have been selected as Cartesian coordinates (x,y,z). However, the radar measures range, azimuth, and elevation. The assumed relationship between the states and the measured values are as shown in Fig. 1 and given by the equations below.

$$x = R \cos\theta \sin\phi \tag{13}$$

$$y = R \cos\theta \cos\phi \tag{14}$$

$$z = R \sin\theta \tag{15}$$



$$x_o = R_o \cos\theta_o \sin\phi_o$$

$$y_o = R_o \cos\theta_o \cos\phi_o$$

$$z_o = R_o \sin\theta_o$$

Figure 1. Illustration of assumed coordinate system used in the radar filters.

where R is the range to the aircraft, θ is the elevation angle, and ϕ is the azimuth angle of the aircraft from North.

Note that this form departs from that as given in (3) since the relationship between the measured variables and the states is a nonlinear one. If the states were directly observable, then (3) would appear as

$$\begin{bmatrix} z1(k) \\ z2(k) \\ z3(k) \end{bmatrix} = H(k)\ X(k) + \begin{bmatrix} v1(k) \\ v2(k) \\ v3(k) \end{bmatrix} \tag{16}$$

where for a sixth order system

$$H(k) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \tag{17}$$

and

$$X(k) = [x(k)\ \dot{x}(k)\ y(k)\ \dot{y}(k)\ z(k)\ \dot{z}(k)]^T. \tag{18}$$

Similarly, for a ninth order system

$$X(k) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \tag{19}$$

and (20)

$$X(k) = [x(k)\ \dot{x}(k)\ \ddot{x}(k)\ y(k)\ \dot{y}(k)\ \ddot{y}(k)\ z(k)\ \dot{z}(k)\ \ddot{z}(k)]^T.$$

Thus, in both cases the equations reduce to

$$\begin{bmatrix} z1(k) \\ z2(k) \\ z3(k) \end{bmatrix} = \begin{bmatrix} x(k) \\ y(k) \\ z(k) \end{bmatrix} + \begin{bmatrix} v1(k) \\ v2(k) \\ v3(k) \end{bmatrix} \tag{21}$$

where v1(k) is the k^{th} component of noise added to the x coordinate, v2(k) is the k^{th} component of noise added to the y coordinate, and v3(k) is the k^{th} component of noise added to the z coordinate. Due to the linearity of this problem, the three coordinate components might be considered as being statistically independent, in which case the R(k) matrix would probably be a diagonal array consisting of the individual coordinate measurement variances which would be constant for all k. For the nonlinear radar problem, the relationships for each k are

$$z1 = (R + n_r)\cos(\theta + n_\theta)\sin(\phi + n_\phi) \tag{22}$$
$$= x + v1$$

$$z2 = (R + n_r)\cos(\theta + n_\theta)\cos(\phi + n_\phi) \tag{23}$$
$$= y + v2$$

$$z3 = (R + n_r)\sin(\theta + n_\theta) \tag{24}$$
$$= z + v3$$

where n_r, n_θ, and n_ϕ are incremental noise disturbances to the true range, elevation and azimuth, respectively. Expanding (22) yields

$$z1 = (R + n_r)[\cos\theta\cos n_\theta - \sin\theta\sin n_\theta] \cdot [\sin\phi\cos n_\phi + \cos\phi\sin n_\phi]. \tag{25}$$

It is assumed that n_θ and n_ϕ are small angle perturbations, and therefore that

$$\cos n_\theta \doteq \cos n_\phi \doteq 1 \tag{26}$$

$$\sin n_\theta \doteq n_\theta \tag{27a}$$

$$\sin n_\phi \doteq n_\phi \tag{27b}$$

(25) can then be written as

$$z1 \doteq (R + n_r)[\cos\theta \sin\phi - n_\theta \sin\theta \sin\phi + n_\phi \cos\theta \cos\phi - n_\theta\, n_\phi \sin\theta \cos\phi] \tag{28}$$

$$\doteq R \cos\theta \sin\phi + v1 \tag{29}$$

where

$$v1 \doteq -Rn_\theta \sin\theta \sin\phi + Rn_\phi \cos\theta \cos\phi - Rn_\theta\, n_\phi \sin\theta \cos\phi + n_r \cos\theta \sin\phi - n_r\, n_\theta \sin\theta \sin\phi + n_r\, n_\phi \cos\theta \cos\phi - n_r\, n_\theta\, n_\phi \sin\theta \cos\phi. \tag{30}$$

Similar developments for the z1 and z2 measurements yields

$$v2 \doteq -Rn_\theta \sin\theta \cos\phi - Rn_\phi \cos\theta \sin\phi + Rn_\theta\, n_\phi \sin\theta \sin\phi + n_r \cos\theta \cos\phi - n_r\, n_\theta \sin\theta \cos\phi - n_r\, n_\phi \cos\theta \sin\phi + n_r\, n_\theta\, n_\phi \sin\theta \sin\phi \tag{31}$$

and

$$v3 \doteq Rn_\theta \cos\theta + n_r \sin\theta + n_r\, n_\theta \sin\theta. \tag{32}$$

Repeating (4), the equation for the covariance of measurement error matrix is

$$R(k) = E\left\{\begin{bmatrix} v1(k) \\ v2(k) \\ v3(k) \end{bmatrix} [v1(k)\ \ v2(k)\ \ v3(k)]\right\}. \tag{33}$$

It is assumed that n_r is much smaller than R. Evaluating the diagonal terms of R(k) yields

$$R(1,1) \doteq R^2\sigma_\theta^2 \sin^2\theta \sin^2\phi + R^2\sigma_\phi^2 \cos^2\theta \cos^2\phi + R^2\sigma_\theta^2\sigma_\phi^2 \sin^2\theta \cos^2\phi + \sigma_r^2 \cos^2\theta \sin^2\phi \tag{34}$$

$$R(2,2) \doteq R^2\sigma_\theta^2 \sin^2\theta \cos^2\phi + R^2\sigma_\phi^2 \cos^2\theta \sin^2\phi$$
$$+ R\sigma_\theta^2\sigma_\phi^2 \sin^2\theta \sin^2\phi + \sigma_r^2 \cos^2\theta \cos^2\phi \tag{35}$$

$$R(3,3) \doteq R^2\sigma_\theta^2 \cos^2\theta + \sigma_r^2 \sin^2\theta. \tag{36}$$

The off-diagonal elements are simply the expected values of the cross product terms, and are computed in the same way yielding

$$R(1,2) \doteq R^2\sigma_\theta^2(1 - \sigma_\theta^2)(\sin^2\theta \sin\phi \cos\phi)$$
$$+ (\sigma_r^2 - R^2\sigma_\phi^2)(\cos^2\theta \sin\phi \cos\phi) \tag{37}$$

$$R(2,3) \doteq (\sigma_r^2 - R^2\sigma_\theta^2)(\sin\theta \cos\theta \cos\phi) \tag{38}$$

$$R(1,3) \doteq (\sigma_r^2 - R^2\sigma_\theta^2)(\sin\theta \cos\theta \sin\phi). \tag{39}$$

Due to the symmetry of the R(k) array, it is also true that

$$R(2,1) = R(1,2) \tag{40}$$

$$R(3,1) = R(1,3) \text{ and} \tag{41}$$

$$R(3,2) = R(2,3). \tag{42}$$

Note that the R(k) matrix is in fact not a constant array, but is one which is state dependent, or rather R, θ, and ϕ dependent.

E. STATE PREDICTION EQUATIONS

The state prediction equations are used to predict ahead from the current estimate to some arbitrary future point in time. Normally this time is that of the next measurement, however, this is not always the case. As a relevant example, in Coarse Guidance, it is required to predict ahead several times between radar samples, due to the long sampling interval

of the radar, and the need to precisely determine times to order command turns.

1. State Prediction in the Linear Case

The analysis which follows will be addressed primarily to the ninth order case. The state vector is given in (20). The strictly linear prediction equations are given by (12) where

$$\phi(k,k-1) = \left[\begin{array}{ccc|ccc|ccc} 1 & T & \frac{T^2}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & T & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 & T & \frac{T^2}{2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & T & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 1 & T & \frac{T^2}{2} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & T \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right] \qquad (43)$$

$$\Gamma(k) = \begin{bmatrix} T^3/6 & 0 & 0 \\ T^2/2 & 0 & 0 \\ T & 0 & 0 \\ 0 & T^3/6 & 0 \\ 0 & T^2/2 & 0 \\ 0 & T & 0 \\ 0 & 0 & T^3/6 \\ 0 & 0 & T^2/2 \\ 0 & 0 & T \end{bmatrix} \qquad (44)$$

and

$$U(k) = [\dot{a}_x(k) \quad \dot{a}_y(k) \quad \dot{a}_z(k)]^T. \tag{45}$$

Note that although in general the ϕ and Γ matrices are functions of k, in this case they are not. This is an arbitrary choice. Normally, T represents the interval between sampling points and is a constant. If one wished to predict ahead by 2T, he could either perform the prediction operation twice in succession, or simply compute a new ϕ and Γ matrix using 2T in place of T. The former technique is used in this study.

The U(k) array represents the deterministic forcing for each of the three dimensions; the units are distance per sec^3 since the forcing is an acceleration rate. (For the sixth order filter U(k) is an acceleration (distance/sec^2).) This must be a function of k since at each time point the control to the aircraft may be different, and in general will be different.

The above linear prediction equations of motion are those which would normally be used in a Kalman Filter. However, it was found that these equations did not yield results of sufficient accuracy. A nonlinear technique was required to obtain precise results. This is derived below, along with the equations for aircraft response to a bank angle command.

2. Aircraft Response to Bank Commands

The aircraft is sent bank angle commands in order to cause heading changes. The flight profile is assumed to be a coordinated turn, in which there is no motion in the vertical plane, and the heading rate change is proportional to the

bank angle, as described in [6]. The relationship is

$$\dot{\psi} = (g/V)\ \phi_a \tag{46}$$

where $\dot{\psi}$ is the heading rate in degrees/sec, g is the earth's gravitational constant (32.2 ft/sec^2), V is the airspeed in ft/sec, and ϕ_a is the actual angle of bank.

The roll transfer function of the aircraft can be approximated by

$$\frac{\phi(s)}{\phi_c(s)} = 1/(s\tau_b + 1) \tag{47}$$

as given in [6], where τ_b is the response time constant. τ_b is a function of the particular aircraft in use. The discretized version of the solution to the differential equation resulting from (47) is

$$\phi(k) = \phi_c(k-1)(1 - e^{-T/\tau_b}) + \phi(k-1)e^{-T/\tau_b} \tag{48}$$

where T is the interval between predictions, ϕ_c is the commanded bank angle, and $\phi(k)$ is the actual bank angle at time T(k). Then the turning rate is

$$\dot{\psi}(k) = (g/V)\ \phi(k). \tag{49}$$

To get the incremental heading angle change over the time T, integrate (49).

$$\Delta\psi(k) = \int_0^T \dot{\psi}\ dt = (g/V) \int_0^T \phi(k)\ dt \tag{50}$$

$$= (g/V) \left[\phi_c(k)t + (\phi(k-1) - \phi_c(k)) \frac{e^{\frac{-t}{\tau_b}}}{\frac{-1}{\tau_b}} \right] \Bigg|_0^T$$

$$= (g/V)[\phi_c(k)T + (\phi(k-1) - \phi_c(k))(\tau_b)(1 - e^{-T/\tau_b})]. \tag{50}$$

Then

$$\psi(k) = \psi(k-1) + \Delta\psi(k). \tag{51}$$

3. <u>State Prediction in the Nonlinear Case</u>

A ninth order filter is assumed for this analysis. The analysis includes effects of wind motion in x and y, and assumes knowledge of the wind components. Since the filter assumes a constant acceleration track, the acceleration prediction equations are

$$\ddot{x}(k/k-1) = \ddot{x}(k-1/k-1) \tag{52a}$$

$$\ddot{y}(k/k-1) = \ddot{y}(k-1/k-1) \tag{52b}$$

$$\ddot{z}(k/k-1) = \ddot{z}(k-1/k-1). \tag{52c}$$

Since the new heading is known in terms of the command bank angle, the predicted velocities are

$$\hat{\dot{x}}(k/k-1) = \hat{V}(k-1)\sin[\psi(k)] + W_x + \hat{\ddot{x}}(k-1/k-1)T \tag{53a}$$

$$\hat{\dot{y}}(k/k-1) = \hat{V}(k-1)\cos[\psi(k)] + W_y + \hat{\ddot{y}}(k-1/k-1)T \tag{53b}$$

$$\hat{\dot{z}}(k/k-1) = \hat{\ddot{z}}(k-1/k-1)T \tag{53c}$$

where $\hat{V}(k-1)$ is the estimated air speed at time $T(k-1)$, and W_x and W_y are the estimated wind components. The predicted position is approximated through the use of numerical integration using the Euler-Maclaurin summation formula, retaining only the first correction term. In general, this formula is given by [7] as

$$\int_{t_o}^{t_n} f(t)dt \simeq T \sum_{i=0}^{n} f_i - \frac{T}{2}(f_o + f_n) - \frac{T^2}{12}(f'_n - f'_o) + \cdots \text{h.o.t.} \tag{54}$$

The equations involving the integrals are

$$\hat{x}(k/k-1) = \hat{x}(k-1/k-1) + \int_{T(k-1)}^{T(k)} \dot{x}(t)\, dt \tag{55a}$$

$$\hat{y}(k/k-1) = \hat{y}(k-1/k-1) + \int_{T(k-1)}^{T(k)} \dot{y}(t)\, dt \tag{55b}$$

$$\hat{z}(k/k-1) = \hat{z}(k-1/k-1) + \int_{T(k-1)}^{T(k)} \dot{z}(t)\, dt\ . \tag{55c}$$

Applying (54) yields

$$\begin{aligned}\hat{x}(k/k-1) &\doteq \hat{x}(k-1/k-1) + \frac{T}{2}\,[\hat{\dot{x}}(k-1/k-1) + \hat{\dot{x}}(k/k-1)]\\ &\quad - \frac{T^2}{12}\,[\hat{\ddot{x}}_t(k/k-1) - \hat{\ddot{x}}_t(k-1/k-1)]\end{aligned} \tag{56a}$$

$$\begin{aligned}\hat{y}(k/k-1) &\doteq \hat{y}(k-1/k-1) + \frac{T}{2}\,[\hat{\dot{y}}(k-1/k-1) + \hat{\dot{y}}(k/k-1)]\\ &\quad - \frac{T^2}{12}\,[\hat{\ddot{y}}_t(k/k-1) - \hat{\ddot{y}}_t(k-1/k-1)]\end{aligned} \tag{56b}$$

$$\hat{z}(k/k-1) \doteq T\ \hat{\dot{z}}(k/k-1) \tag{56c}$$

where $\ddot{x}_t$ and $\ddot{y}_t$ represent total accelerations in x and y. The simplification of the equation used to predict altitude is a result of the fact that there is no deterministic forcing in that direction.

A discussion of the relationship between the acceleration estimate $\hat{\ddot{X}}(k/k-1)$ and the total accelerations shown in (56a) and (56b) is required at this point. The purpose of estimating an acceleration is due to the fact that bias bank angles exist which tend to cause a continuous turning motion,

and thus additional and unknown accelerations in the x and y directions. The prediction method as described above will place the aircraft at exactly the correct point with the correct velocity if there is no bias. In this case, the estimator will estimate that zero additional acceleration is present. In the case in which an unknown bias bank angle is present, the prediction equations will not be moving the aircraft heading angle the correct number of degrees/sec and thus lag will develop in the filter. This lag will cause a non-zero acceleration to be estimated, and this value will subsequently be added appropriately to the respective velocity components in order to compensate for the insufficient motion.

In summary, the acceleration estimate is not an estimate of the total acceleration, but is only an estimate of that acceleration which is unknown, due possibly to a variety of causes, but due primarily to an unknown bias angle in the autopilot roll reference. Thus the derivatives with respect to time of the velocity components in (56a) and (56b) do not equate to the acceleration estimates. Rather, they may be computed by differentiating equations (53) with respect to time and assuming that $\hat{\ddot{X}}(k-1/k-1)$ is approximately constant. This yields

$$\hat{\ddot{x}}_t(k/k-1) = \hat{V}(k-1)\ \cos[\psi(k)]\dot{\psi}(k) \tag{57a}$$

$$\hat{\ddot{y}}_t(k/k-1) = -\hat{V}(k-1)\ \sin[\psi(k)]\dot{\psi}(k) \tag{57b}$$

and

$$\hat{\ddot{x}}_t(k-1/k-1) = \hat{V}(k-1)\cos[\psi(k-1)]\,\dot{\psi}(k-1) \tag{58a}$$

$$\hat{\ddot{y}}_t(k-1/k-1) = -\hat{V}(k-1)\sin[\psi(k-1)]\,\dot{\psi}(k-1). \tag{58b}$$

F. KALMAN FILTER IMPLEMENTATION

Two separate Kalman filter subroutines were developed to simulate the operation and filtering of the radar processor. RADAR6 is the sixth order filter and RADAR9 is the ninth order filter. Their implementation in software is very similar; the differences are described below along with specific characteristics of the Kalman filter which apply to both versions. A general flow diagram for the filter is presented as Fig. 2.

1. Differences in RADAR6 and RADAR9

The obvious difference between the two filters is that one is capable of tracking a maneuvering aircraft with nonzero autopilot bank angle bias, and the other is not. This is due to the fact that the sixth order filter does not include the acceleration states in x, y, and z. The price paid for this additional estimation capability is an increase in computation time and program size. As implemented, the ninth order filter requires about 32 percent more storage allocation in memory than does RADAR6; RADAR6 requires about 76 K bytes of storage on an IBM 360-67. The increased computational time is difficult to judge since all timing data refers to the running of the overall program. The estimate for increased run time for the ninth order filter is on the order of 20 to 50 percent, depending on overall program run time.

The increased run time and storage requirements result from the requirement that the ninth order filter must have

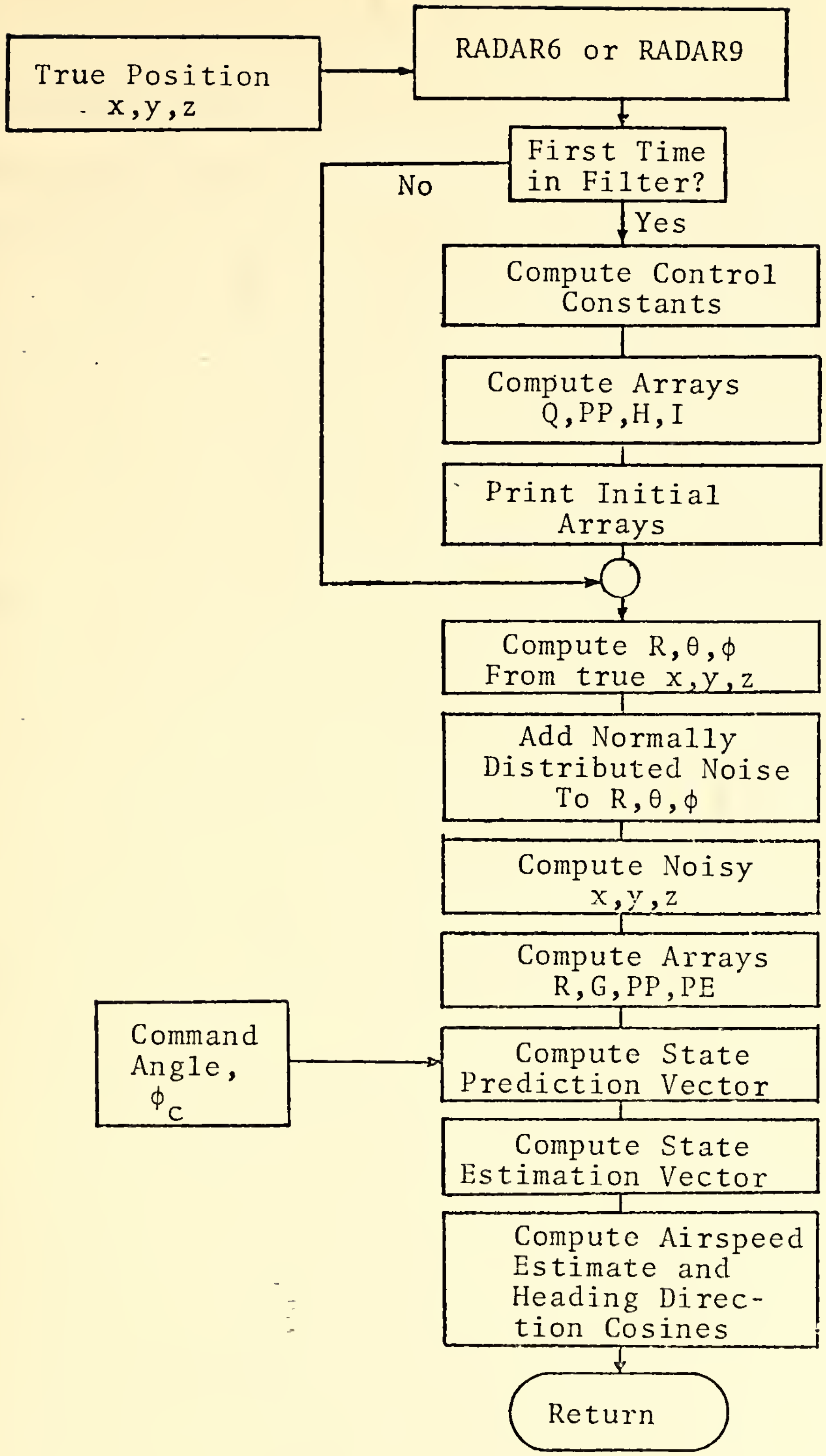


Figure 2. General block diagram of Radar Subroutines.

arrays which are 9 x 9, while the sixth order filter requires only arrays which are 6 x 6; 18 of these arrays are involved resulting in a factor of 2.25 increase in array storage alone.

The prediction calculations for the sixth order filter differ from the ninth order filter only in the fact that the acceleration terms are missing in augmenting the predicted velocity components.

RADAR6 was used to develop the Coarse Guidance program since no mention of autopilot bias problems was found in the available documentation on that original system, [8] and [9]. RADAR9 was developed directly in response to the bias problem and was therefore used exclusively in the improvement of the Precision Guidance simulation program. The two subroutines could be interchanged to work with the other simulation mode's program in a matter of minutes, should this be desired.

2. Initialization and Constant Array Calculations

Upon entering either RADAR subroutine for the first time, logic passes control through a section of constant array calculation and definition of program constants. The constants specified at this point are primarily those which are used in bank and turn angle equations. These are a function of the aircraft type and update intervals and remain constant throughout a given simulation run.

A total of five arrays are defined at the program start and remain constant. An Identity array of order equal to the filter order is set up. The Measurement matrix, (H),

as given in either (17) or (19), depending on filter order is then defined. The state transition array, ϕ, is considered constant for this application and is defined in accordance with the filter order. The Q matrix, which is a measure of the expected unknown random forcing to be applied to the system is computed using the Γ array as given in (44) (for the ninth order filter) in conjunction with the expected value of the random forcing array, W. For this study, W has been set to $\underline{0}$. The use and effects of non-zero values in this array are discussed below.

The covariance of the initial state prediction vector is set to 10^6, a somewhat arbitrarily large number. The effect of choosing such a large variance on the states is to cause the filter to set the initial state estimation vector equal to the observation. In other words, the filter essentially ignores the *a priori* information set in as initial conditions. This is a very typical method of initializing a linear Kalman filter, when little confidence is placed on any initial estimate of the states. The original simulation programs followed this practice (of zero initialization) and it was continued in the improved version. However, considerable improvement in initial filter settling for the Kalman filters implemented could be achieved through the use of good initial conditions. The improved version of Coarse Guidance yields good estimates of the aircraft position, and in a "pass-off" to the Precision Guidance mode in a "live run" the states active in Coarse Guidance at the time would make

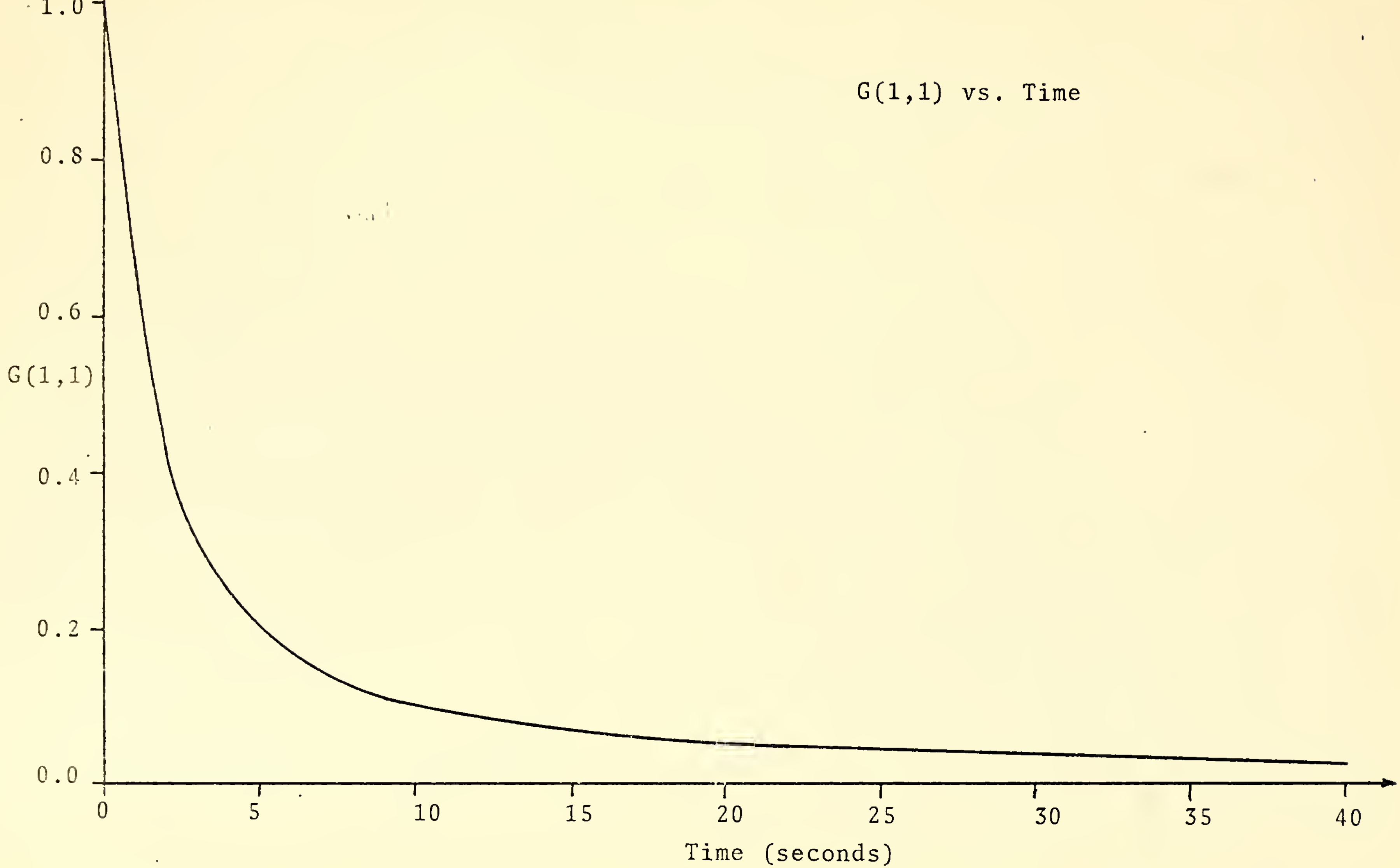
valid initial conditions for the Precision Guidance radar filter.

G. PREDICTION ERROR, THE GAIN SCHEDULE, AND COVARIANCE OF AIRCRAFT MANEUVER

The gain matrix, G(k), determines to what extent the data will be permitted to affect the state estimation. During the initial few estimations, the data usually always plays a large role in determining states, due to the large uncertainties in the actual states specified by the large covariances of prediction and estimation. As more data is taken, the covariances fall, and there is less requirement to accept noisy data into the filter to update the state estimate. Figure 3 shows a plot of the typical gain schedule for the x coordinate, G(1,1) term as a function of time. Values for the primary gain terms in y and z are similar. The curve is approximately exponential in shape, starts essentially at 1.0 at time zero, and decreases monotonically. As can be seen from (9), the exact values in the gain matrix depend on a multitude of variables, including the states themselves in the form of the R matrix.

Absolute compliance with the Kalman filter assumptions results in a gain schedule which is optimal in the sense that the state estimates will be of minimum variance. The use of changing gains with time is implicit. Note that in the original simulation programs, only constant gain filters were used, or in a few instances, filters which started at one constant gain value and then switched to one different set of gains.

Figure 3. Illustration of an actual gain schedule decreasing as a function of time.



The gains are related to the filter bandwidth. High gains correspond to wide bandwidth since they "let in" nearly all of the measured data, including the noise on the data. Lower gains correspond to narrow band filters since very little data gets into the state estimation calculations. Use of low gains also resembles a narrow band low pass filter in the phase lag which results in the state estimation vector due to an abrupt change in the actual states. When this results, large differences begin to develop in the prediction residual of the estimation equation (11):

$$E(k) = Z(k) - H(k)\hat{X}(k/k-1). \tag{59}$$

In this case, (59) simplifies to

$$E_x = x_{data} - \hat{x}(k/k-1) \tag{60a}$$

$$E_y = y_{data} - \hat{y}(k/k-1) \tag{60b}$$

$$E_z = z_{data} - \hat{z}(k/k-1). \tag{60c}$$

As the differences in equations (60) begin to be biased either positively or negatively over a period of time, the filter will begin to integrate to a new trajectory in state space to compensate for the fact. The lower the gains, the longer this process will take.

If no random excitation noise, or no unknown forces or uncertainties are present in the system whose states are to be estimated, then the proper setting for W, the covariance of random state excitation, is $\underline{0}$. Since the Q array in (9) is

$$Q(k) = \Gamma\, E[W\, W^T]\Gamma^T, \tag{61}$$

if W is $\underline{0}$, then Q will also be $\underline{0}$. Examination of (9) in this case will reveal that as $t \to \infty$ the gain schedule will go to zero, and as a result data will have little effect on the state estimation process after a relatively short period of time after initialization. This is fine if all factors are known.

In real systems, all factors affecting the states are rarely known. Uncertainties in AN/TPQ-27 might include wind velocity, exact aircraft roll response, biases in measurement equipment, and time variations of all of these.

The goal in establishing improved simulation programs was to devise techniques which would perform the required estimation and tracking assuming the above uncertainties did not exist to a significant extent. This attitude seemed to follow that used in the generation of the original program. It was recognized that use of a gain schedule which goes to zero is not a likely useable solution to the problem; however, it did seem reasonable to try to generate a solution which worked with zero Q when the uncertainties did not exist. With this accomplished, actual utilization of the algorithms against real data will determine the extent to which the bandwidth must be opened to achieve the best results.

Selection of Q is a problem which perpetually plagues users of the Kalman filter. Any given data set will have a given Q which will yield best results in terms of a specified performance criteria, normally tracking precision. The Q cannot be selected with the benefit of hindsight, and must be chosen to yield the optimum performance over the ensemble of state trajectories of interest. The normal method for finding

this value would be to take as many raw data tracks as possible, and try to determine the Q using statistical methods.

An alternate technique to generate Q on-line through the use of prediction error is described in [10]. In this paper, Aldrick and Krabill propose the calculation of Q by the following method:

$$Q(k) = a[\mathrm{Del}(k)\ \mathrm{Del}(k)^T] + b[\mathrm{Del}(k-1)\ \mathrm{Del}(k-1)^T] \quad (62)$$

where a and b are constants to be determined by analysis of actual data, and

$$\mathrm{Del}(k) = \hat{X}(k/k) - \hat{X}(k/k-1)\ . \quad (63)$$

This method was investigated to a limited extent, and showed some promise if refined. Simple use of (62) seems to open the bandwidth wider than is desired. It has the advantage over the use of some constant Q for all runs that, in theory, if no uncertainties exist, the Del(k) arrays will be zero and thus Q will be $\underline{0}$. Thus, wide bandwidth is achieved only when required, as determined by prediction success. In practice however, it was found that this method caused the gains to oscillate, and created excessive error.

H. REDUCTION OF COMPUTATIONAL TIME

The most obvious disadvantage of using the Kalman filter compared with the constant gain alpha-beta filter is the increase in complexity and computation time required to compute this optimal gain schedule. There are several techniques which can be employed to reduce this burden, all of which result in further loss of optimality, but to limited extents.

In general, the R matrix is a function of the states or position of the aircraft. If the rough start and end points for the tracks to be followed are known in advance of the mission, it is possible to compute the average R, θ, and ϕ and thus an average R matrix. Unless R, θ, and ϕ vary over wide ranges, the loss in optimality by this approximation should be small. Since the gain schedule in state dependent only to the extent that R is state dependent, the gain schedule can now be computed and stored is a mission data table, and thus need not be computed on-line. Since 18 different gains are required by RADAR6 and 27 by RADAR9 for each sampling time, this becomes a rather large problem if auxillary storage is not available.

An alternative to storing the gain schedule is to fit each of the gain schedules to either an exponential curve or to a function of the type

$$f(x) = a + b\,x^{-1} + c\,x^{-2} + d\,x^{-3} + \cdots \qquad (64)$$

Then only 18 (or 27) computations would be required at each sampling time.

Whatever technique is chosen to approximate the true gain schedule, it will not be nearly as sub-optimal as a constant gain over all time, or two constants which are switched in and out.

III. PRECISION GUIDANCE SIMULATION

A. INTRODUCTION

A limited amount of Precision Guidance simulation program documentation was provided with the original version of the program. This greatly facilitated understanding of the original version of the program, and also subsequent modification for improvement. Many of the original concepts used were not changed. Input and output formats were revised for ease of use and to provide a greater indication of program performance. Since large portions of the original program's logic and coding have been retained, these may be mentioned and summarized lightly for continuity. The derivations and justification of assumptions in these areas will not be addressed, as they are considered to be adequately documented in the existing system documentation.

A block diagram of the simulation program control loop is presented in Fig. 4. In a few words, the program starts with the aircraft at some designated position and velocity with respect to a target which is to be bombed. After a 6 sec filter settling period, control logic determines the lateral error which will be incurred if the present flight profile is continued until time to release the bomb. Functions of the lateral error become the controlling mechanism to generate command bank angles which cause the aircraft to change course. It is emphasized that all control commands

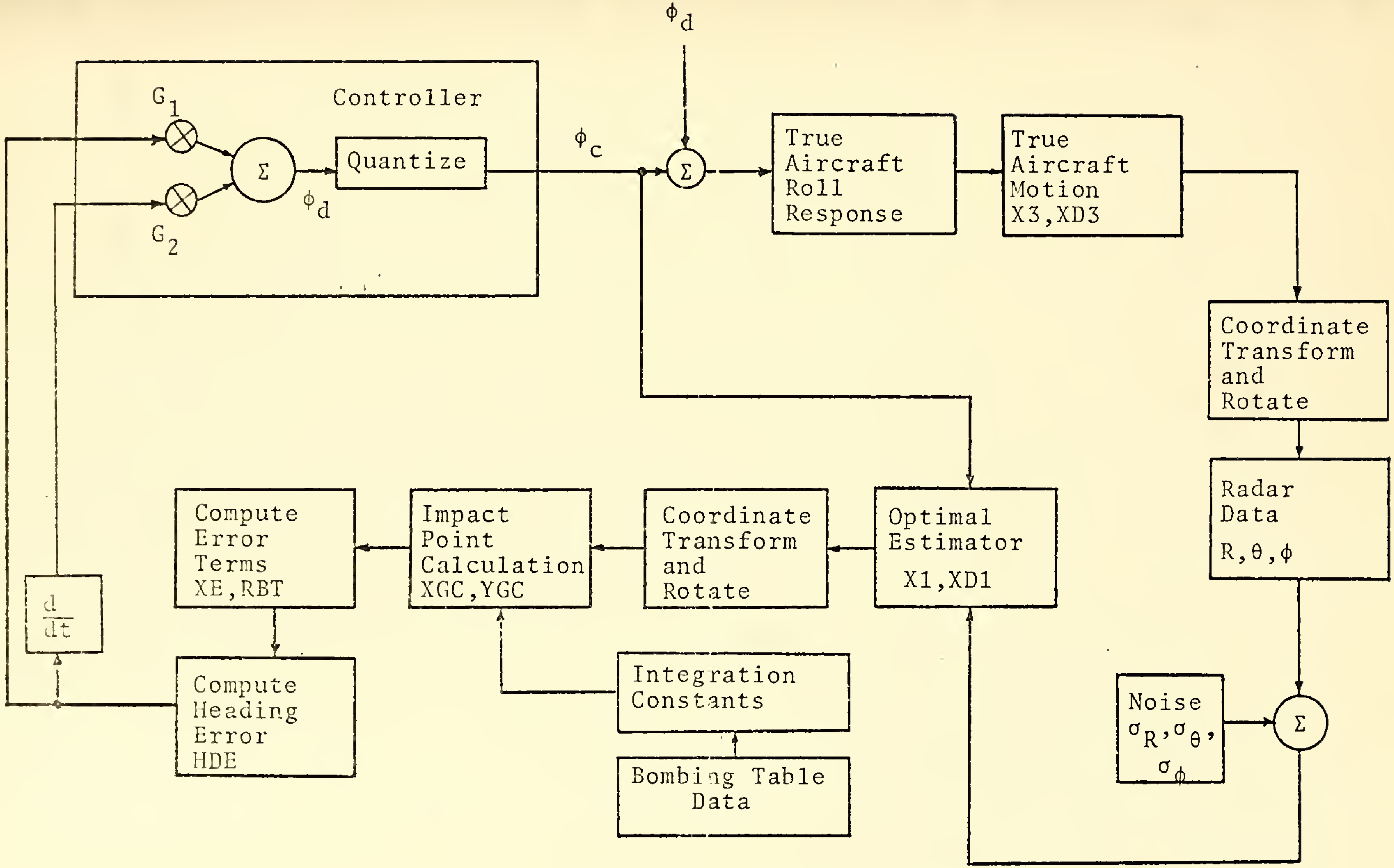


Figure 4. Simulation Program block diagram.

are based on estimated positions and velocities. Thus it may be possible to drive the estimated lateral error to zero, and still miss the target considerably, if the state estimation vector is in error. One further comment is that the problem is not totally one of estimation and control. The aircraft has a bank angle limitation imposed of ± 30 degrees. This limits the turning rate such that, depending on the initial position and velocity with respect to the target, the aircraft may not be physically able to come about to the correct heading in the time required. Examples of this situation are provided.

B. INITIALIZATION AND FILTER SETTLING TIME

The new simulation program is functionally similar to the original version in initialization. The same variables are used wherever meaningful. The coordinate transformation vectors and matrices defined during initialization are retained, as are all equations for operating in the dive bombing mode. (The original program provided logic for executing a diving mode, but use of this mode has not been investigated in this study.)

The precision radar sampling rate is 8 Hz. No action except state estimation is taken during the first 2 sec of the simulation run. At the 2 sec point, logic is executed to enable prediction of the lateral error at 6 seconds into the run. The 4 second lag is assumed to be due to an attempt to simulate the fact that the actual computer used is compute-time limited, thus constraining the integration logic to

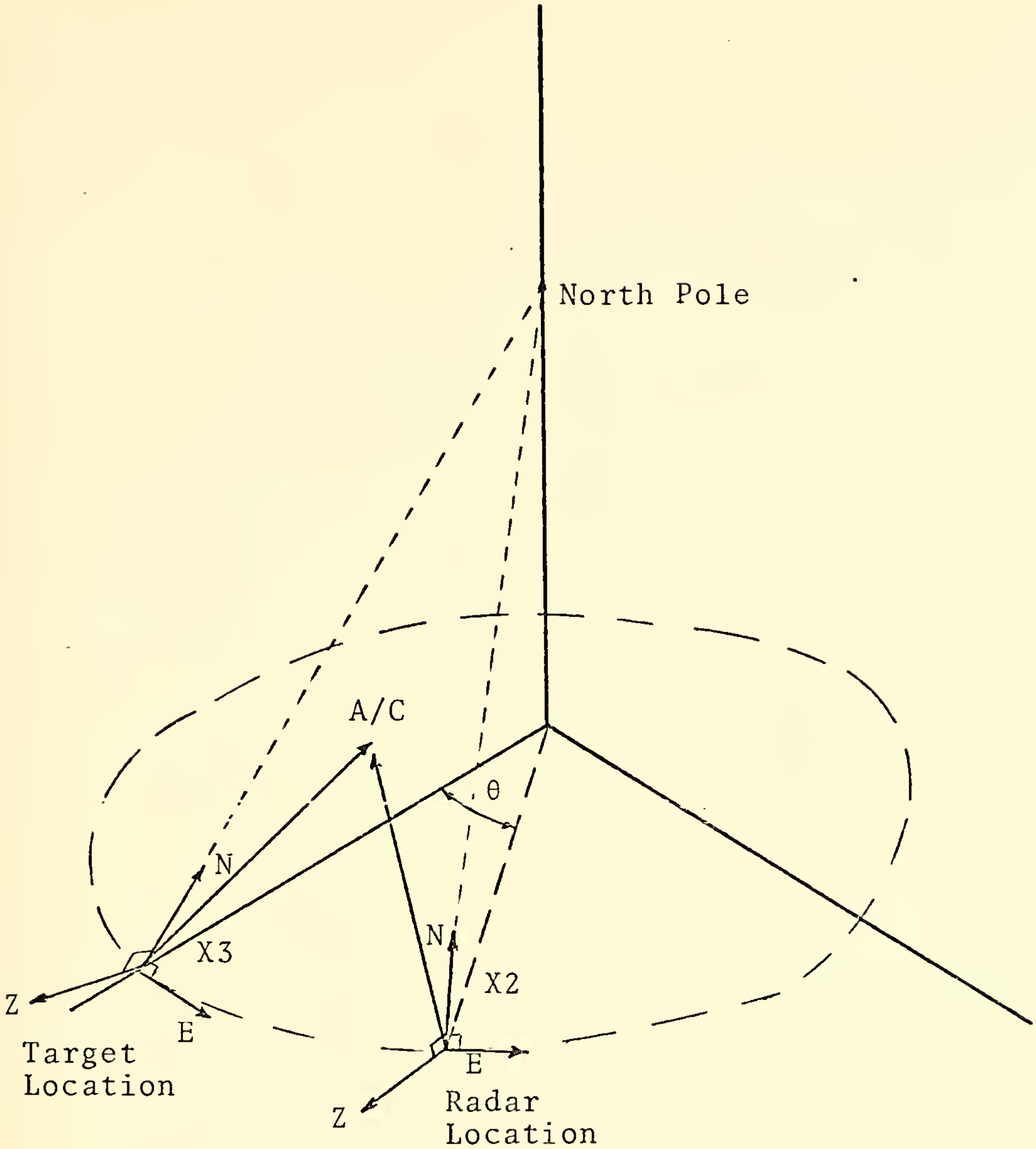
execution at no more than 0.25 Hz. At t = 6 seconds, the lateral error is first estimated, and the first non-zero command to the aircraft can be transmitted.

C. PRECISION GUIDANCE COORDINATE CONVENTIONS AND TRANSFORMATIONS

Three primary coordinate systems are used in Precision Guidance calculations. These systems have their origins at the target, the radar, and the aircraft. A convention in notation is used throughout the program to denote vectors in the various coordinate systems. X1 represents a position vector, XD1 represents a velocity vector, and XDD1 represents an acceleration vector. (The only time acceleration vectors appear are in RADAR9.) The number "1" in the notation simply refers to a specific coordinate system.

X1 represents the state estimate of the aircraft in the radar reference frame; X2 represents the true state vector of the aircraft in the radar frame. X3 represents the state vector of the aircraft in the target frame. Conversion from one frame to the other is accomplished through the use of transformation matrices EM1, EM2, and EV1, and the subroutines MATMLT, and MATMAD which perform the matrix multiplication and addition. The relationship between X2 (or X1) and X3 is illustrated in Fig. 5.

Aircraft control is derived through the use of the X6 coordinate system, which has its origin at the aircraft and its y axis oriented along the estimated aircraft ground heading. This transformation is accomplished through the coordinate transformation matrix EM3. Figure 6 illustrates the



The local vertical is labeled Z. Local North is the y axis, and local East is the x axis in each frame. The dashed line is a great circle through the two points. Point labeled A/C is the aircraft.

Figure 5. Illustration of the relationship between the X3 and the X2 coordinate systems.

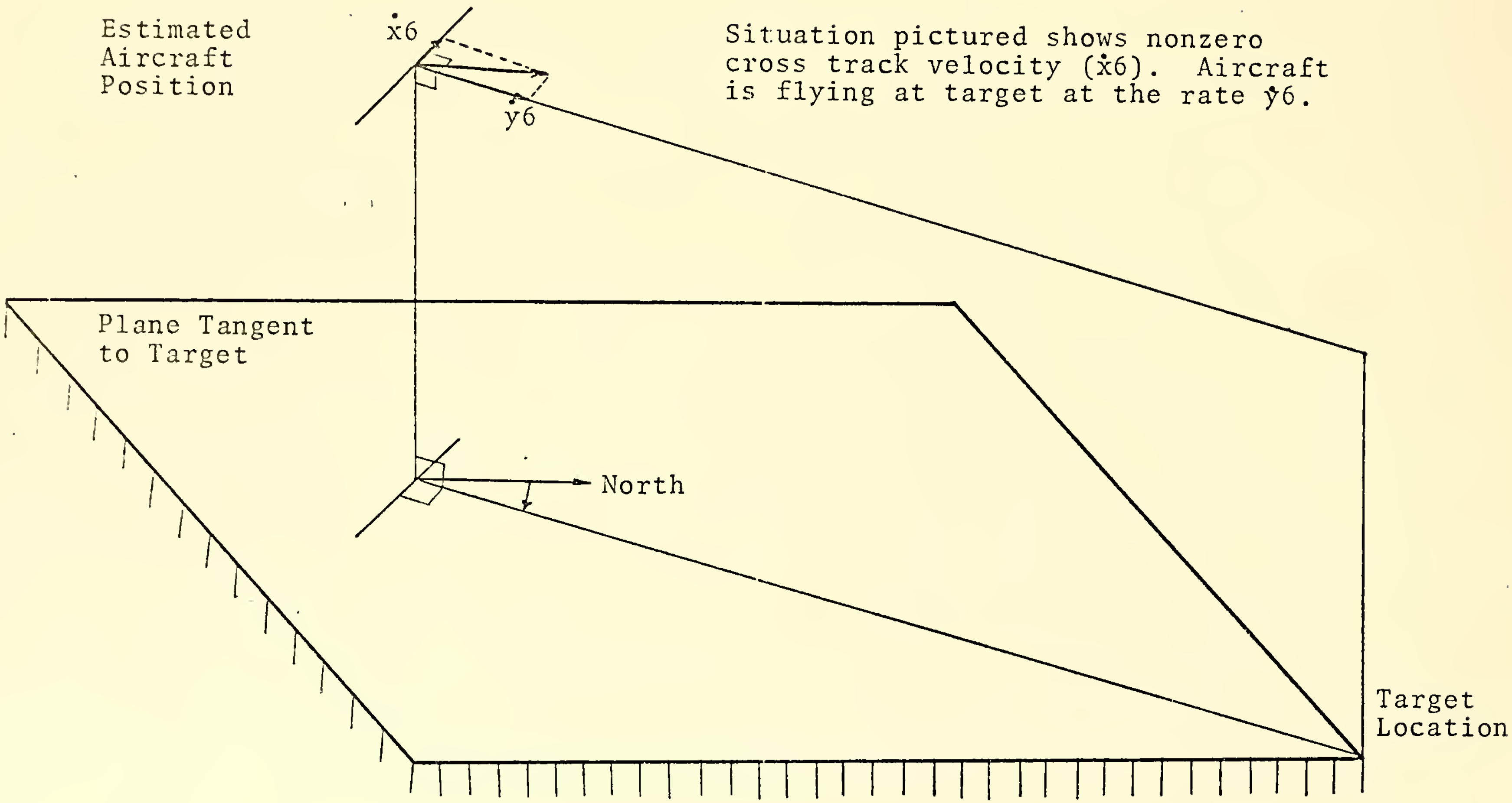


Figure 6. Illustration of relationship between X6 and the Target.

the relationship between this coordinate system and the target. Note that in the case of perfect estimation and no wind that this y axis orientation would point directly at the target.

Each of the X, XD, and XDD vectors are of dimension 3, one storage allocation for each physical dimension. Thus,

$$X1(1) = x1$$

$$X1(2) = y1$$

$$X1(3) = z1.$$

The coordinate systems and transformation equations are fully specified in [11].

D. LOOP GEOMETRY AND ERROR CALCULATIONS

The normal flow of control through the main program loop following the initial 6 second settling period begins with a time update and a movement of the aircraft in accordance with the command bank angle generated at the previous time, all within the original subroutine ARCRFT. The true position is then transformed into the radar frame, noise added, and the new state vector estimated using the RADAR9 subroutine. The estimated state vector is then transformed into the X6,XD6 system, sometimes referred to as the "double primed" reference system in the original program documentation.

As mentioned previously, every 4 seconds the simulation program computes new coefficients which are used to calculate the ballistic range to the target, and bomb time of fall. This involves integration of a system of 16 differential equations using a fourth order Runge-Kutta scheme. The ballistic

range (RA) and time of fall for the bomb (TF) are computed using a first order linearized approximation to the system of equations described above at those times when no integration has been performed. Subroutines STIFF, DER, and OUT are used to perform the required calculations; these are completely specified in [10].

The ballistic range, RA, is the distance the bomb will impact from the present aircraft position. Time of fall, TF, is the time in seconds which will elapse between bomb release and impact. These values and the ballistic wind components in track and cross-track to the aircraft estimated heading determine the impact point for the bomb in X6 reference frame. The equations for this estimated impact point, (XGC,YGC), are as described in [11]. Figure 7 illustrates the geometry of the situation for the windless case. It is assumed that by selecting the bomb release time carefully, a near-zero error in impact on the y6 axis can be achieved. Only by having the aircraft heading precisely correct will the cross track impact error, called lateral error (XE), be zero as well.

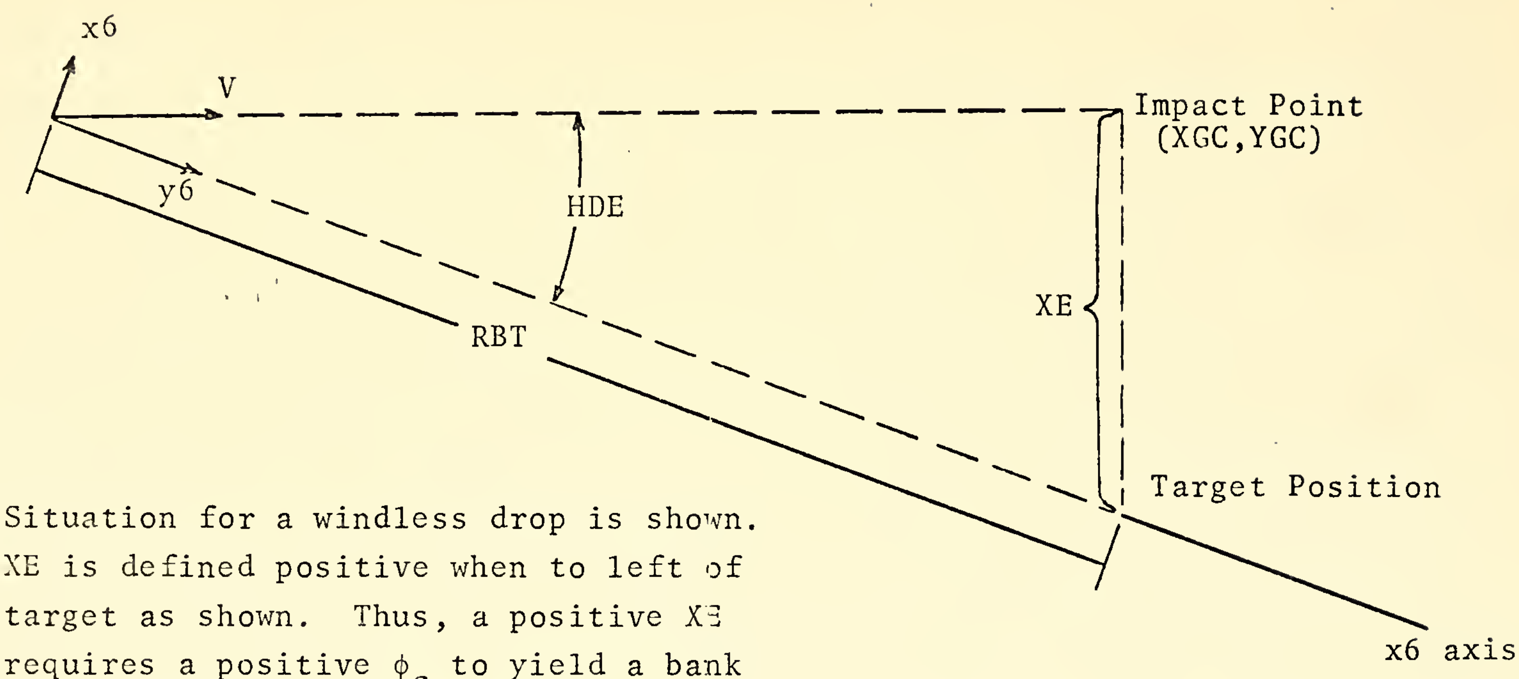
At each sampling point, the Time-to-Go-to-Release the bomb, TG, is computed using

$$TG = (y6 - YGC)/\dot{y}6 \ . \qquad (65)$$

The lateral error is given by

$$XE = x6 - XGC.$$

Other quantities which are used in error calculations are the ballistic range to the target, RBT, and the heading angle error, HDE. These are given by



Situation for a windless drop is shown. XE is defined positive when to left of target as shown. Thus, a positive XE requires a positive ϕ_c to yield a bank to the right.

Figure 7. Illustration of error and bombing geometry.

$$RBT = (x5^2 + y5^2)^{\frac{1}{2}} \tag{67}$$

where X5 is the coordinate system X6 prior to being rotated for heading-target alignment, and

$$HDE = \arcsin(XE/RBT). \tag{68}$$

E. AIRCRAFT CONTROLLER DESIGN

The new controller for the aircraft is considerably less complex than that originally used. The original version employed lead-lag networks, suitably digitized, with constants which were switched in or out at different stages of the simulation run. The lateral error was driven to zero in the original version of the program, by selection of a gain constant times the lateral error to yield a desired bank angle.

The new controller design attempts to drive both an error and an error rate to zero. The error signals to be driven to zero are the heading angle error, HDE, and the heading angle error rate, HDEDOT, where

$$HDEDOT = \dot{HDE}. \tag{69}$$

The desired control bank angle, ϕ_d, is given by

$$\phi_{d1} = G_1 \text{ HDE} \tag{70}$$

$$\phi_{d2} = G_2 \text{ HDEDOT} \tag{71}$$

and

$$\phi_d = \phi_{d1} + \phi_{d2}. \tag{72}$$

This procedure requires selection of the feedback gain constants G_1 and G_2. It should be noted at this time that G_1 and G_2 are the only two constants in the simulation program which must be determined "through simulation," i.e., by

trial and error. The original program was filled with numerous "gain constants" which were, or were to have been selected by "simulation." In addition, in the original version, much of the theory was developed through the assumption that the process was approximately a linear one, and development of a linearized model which reflected these linearizations. Such linearization may be somewhat valid in the final stages of a long run where only small commands are being sent, but in the initial phase of heading correction, the commands saturate and the assumption of linearity is invalid. It is in this nonlinear region of operation that commands must be optimized to yield a combination of rapid correction of heading error and minimum overshoot of the correct heading.

Two sets of control gains have been selected for the two prevalent aircraft roll response time constants, $\tau_b = 2$ and $\tau_b = 3.3$. The gain constants are given below.

$$\tau_b = 2 : \quad G_1 = G_2 = 75$$

$$\tau_b = 3.3 : G_1 = G_2 = 150.$$

These parameters were selected through analysis of numerous simulation runs with each τ_b and subjectively evaluating the resultant performance. The performance criteria used was to generate initial commands which are of the "bang-bang" type referred to in optimal control theory, causing the bank command to saturate at the autopilot limit of ± 30 degrees. This causes the aircraft to begin turning toward the correct heading at the maximum possible rate permitted. The effect

of heading error rate control is to act as a damper in that the faster the heading angle changes, the more the control will begin to decrease. The second part of the performance criteria was to require that when small angle errors are noted, only small commands are generated.

There is a requirement for no bank commands during the final second of flight. This is to ensure that the aircraft does not make a violent maneuver just as the bomb is dropped, causing the bomb to be pitched from the aircraft rather than simply dropped as had been assumed in the bombing calculations.

For a one second period beginning three seconds before bomb release, i.e., TG = 3 to TG = 2, the average estimated aircraft bank angle is computed. This average estimated bank angle is then sent as the command during the final two seconds before bomb release. The reason for sending this value vice a command of zero degrees is the possible presence of a bias angle. Figure 8 is a plot of a typical simulation run in which the autopilot has a bias bank angle of 3 degrees. Note that this bias is automatically compensated for by the commands sent. The average command at the end of the run is approximately -3 degrees. This has the effect of rolling the aircraft back to a level flight profile in a steady-state situation, vice a slowly turning profile which would be present if the command was 0 degrees. By averaging the estimated bank angle over the period defined above, and sending this value for the final few seconds, the aircraft continues on an approximately error free path. A zero command

ROLL ANGLE(ACTUAL AND ESTIMATED) VS. T

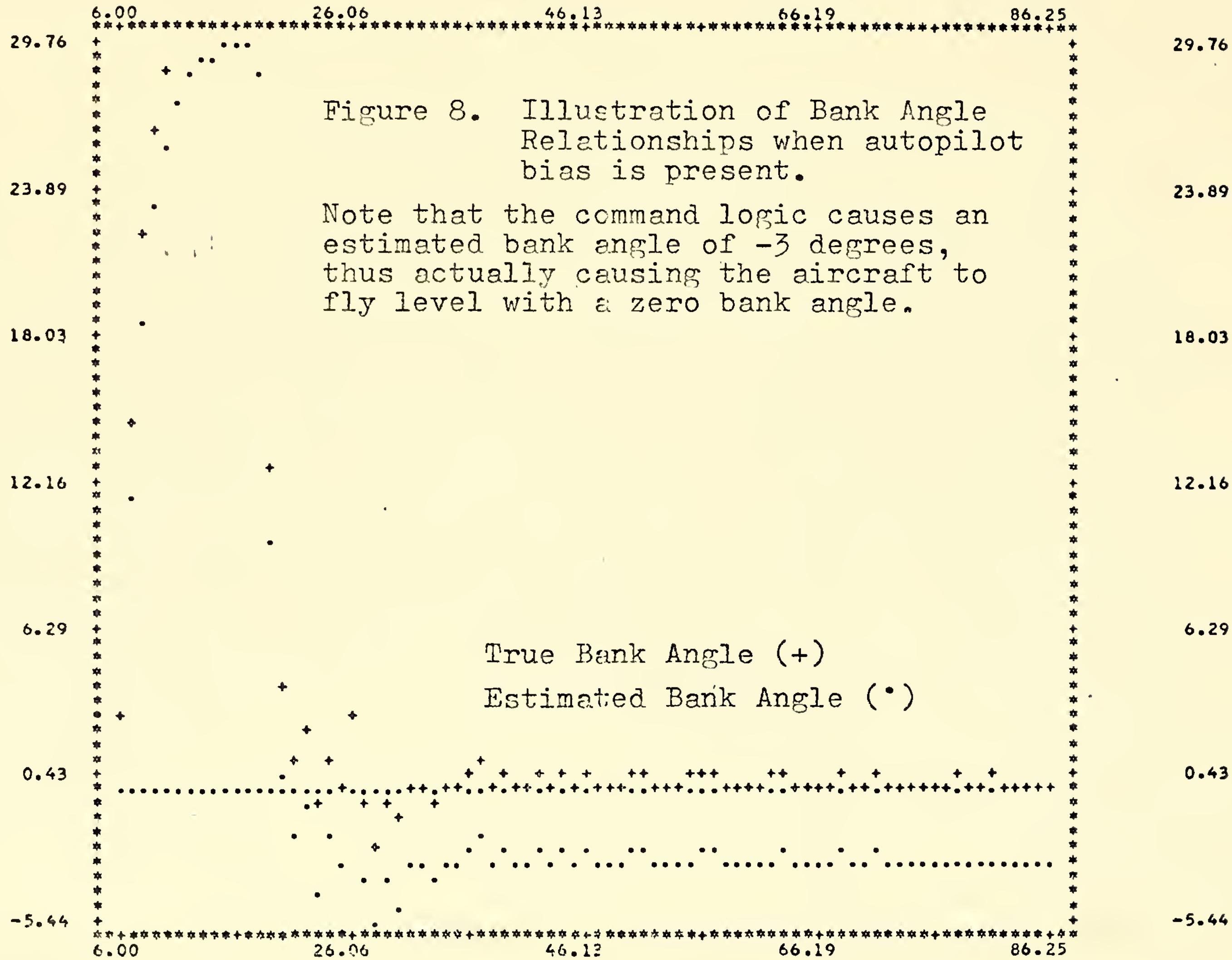


Figure 8. Illustration of Bank Angle Relationships when autopilot bias is present.

Note that the command logic causes an estimated bank angle of -3 degrees, thus actually causing the aircraft to fly level with a zero bank angle.

at this time would cause the aircraft to begin a slow turn and greatly increase the lateral error with no opportunity for correction.

Command system requirements require that the command angle be quantized to the nearest 15/128 degree. This is accomplished through simple logic as described in [11].

F. PRECISION GUIDANCE PROGRAM IMPLEMENTATION

1. Main Program and Subroutines

The Precision Guidance simulation program consists of a main routine and seven primary subroutines. The program listing is appended at the end of this report. Each of the subroutines is included and in each of the routines the primary variables and purpose of the routine is specified, with the exception of subroutines STIFF, DER, and OUT, which serve only to support the Runge-Kutta integration defined above. The simulation program was written to run on an IBM 360-67 computer system and takes advantage of several of the system software subroutines. The program with linkage and subroutines requires approximately 150 K to execute and will run a typical time of from one to three minutes of CPU time. Compilation requires approximately 50 seconds.

A listing of the variables used in the program along with the variable definitions is provided in Appendix A.

2. Program Input and Output

Specific format definitions for program input data are provided in Appendix B. The program output has been

changed significantly from that in the original program version. A sample of one complete simulation run is provided at the end of this report. The initial conditions and constants are output along with short word definitions of their meaning or useage. Each second, a summary of the critical parameters of the program are output in block form; the output key is printed on each page for ready availability to the user. During the final second before bomb release, the critical parameters summary is printed at every sampling point.

On the final time through the main processing loop, all estimates are replaced by their corresponding true values, and the bomb "released." The impact point is computed, and the miss distances in x,y and overall are printed, designated XI, YI, and RI, respectively.

As a measure of filter effectiveness in noise reduction and state prediction in position including deterministic motion, the square of filter residual is summed throughout the run. The RMS values of filter residue are printed following the bomb impact miss distances. Filter residue in this sense is defined as

$$FILRES_x = x1 - x2 \tag{73a}$$

$$FILRES_y = y1 - y2 \tag{73b}$$

$$FILRES_z = z1 - z2. \tag{73c}$$

In addition to the filter residual in each coordinate, a radial residual defined as

$$FILRES_R = (FILRES_x{}^2 + FILRES_y{}^2 + FILRES_z{}^2)^{\frac{1}{2}} \qquad (74)$$

is printed in RMS form, where the average is over the entire simulation run.

Some of the critical parameters are stored in arrays each second throughout the run. These are then line printer plotted at the completion of the run with appropriate labelling.

IV. COARSE GUIDANCE SIMULATION

A. INTRODUCTION

Two documents, [8] and [9], were provided to describe the techniques and general approach being followed on the original Coarse Guidance simulation program. However, no program documentation such as that provided for Precision Guidance, [11], was available. The program provided seemed inordinately complex in some places and the general approach to the problem did not appear to be a viable one from which to build an improved version.

It was determined that the best approach would be to write a totally new main simulation driving routine, use the already existing aircraft simulation program, ARCRFT, and the already developed Kalman filter radar simulation program RADAR6 to simulate the Coarse Guidance tracking system. RADAR6 was chosen vice RADAR9 simply because neither [8] nor [9] mentioned any difficulty with bank angle biasing. Also, the smaller size and faster running time of RADAR6 made that subroutine a preferred choice. The decision to mate RADAR6 to Coarse Guidance rather than RADAR9 is by no means a final choice. A matter of only a few minutes would be required to modify RADAR9 to be compatible with the Coarse Guidance simulation program. Thus if biasing is a problem in this mode also, RADAR9 could serve as the appropriate unbiased state estimator. The changes in RADAR6 from RADAR9 other than those which relate to acceleration estimation will be discussed

below. A few simplifying changes in the ARCRFT subroutine were also accomplished to reduce core storage requirements and execution time; these will also be described.

Some of the concepts and a few equations from the original documents on Coarse Guidance were used in this study. Since the new simulation program differs significantly from the original version, nearly all equations will be presented and most will be derived.

The basic concept in Coarse Guidance is to get the aircraft from some initial starting point to the final bombing run by flying a predetermined course which is specified by "waypoints" and azimuths of course "legs." A typical simulation setup might appear as illustrated in Fig. 9. The designated legs presumably follow a "safe" path for the strike aircraft. Also, presumably, if the aircraft deviate from the specified path too far, they become in danger. Therefore, it is desirable to provide some control to keep the aircraft as near to the specified path as possible. Of particular importance is to recognize the approach of the beginning of a new leg and begin a "command turn" onto this new leg at such a time that upon completion of the turn, the aircraft will be on the new leg with the same ground heading as the leg's azimuth.

B. PRE-MISSION DATA TABLE COMPUTATIONS AND INITIAL CONDITIONS

Once the path to be flown is specified, the individual legs can be characterized by their azimuth with respect to North and their length. The beginning of the mission is

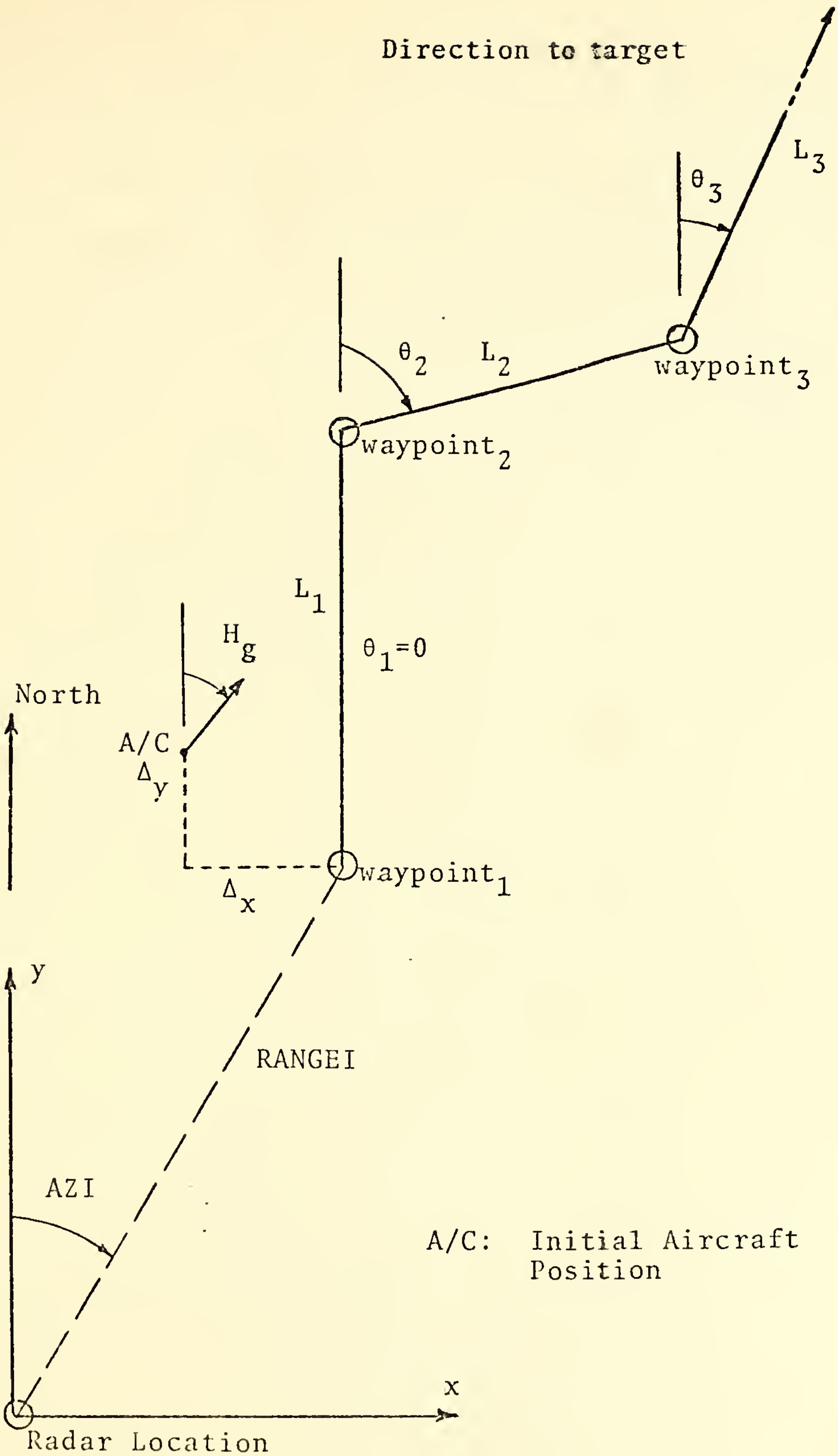


Figure 9. Illustration of a typical 3 leg course.

referred to as a TACAN Entry Point, and is the approximate position of aircraft entry into the problem. Wind causes the ground heading and the air heading to differ. Since most of the aircraft control calculations are made with respect to the air heading, effects of wind must be considered. The initial position of the aircraft is placed at the beginning of the first leg perturbed by some error, and with ground heading of the first leg's azimuth also perturbed by some angle deviation from desired.

1. <u>True and Estimated Wind Components</u>

Figure 10 illustrates the relationship assumed for wind in the problem. Provision is made for an error in estimated wind speed and direction. All true aircraft motion is computed using true wind. All estimated aircraft motion and control decisions are made using the estimated wind components. Wind is assumed zero in the vertical direction. If the true (estimated) direction toward which the wind is blowing is θ_w (θ_{wh}), and the true (estimated) wind speed is V_w (V_{wh}) then the components of wind are given by

$$w_x = V_w \sin(\theta_w) \tag{75a}$$

$$w_y = V_w \cos(\theta_w) \tag{75b}$$

for the true wind, and

$$w_{xh} = V_{wh} \sin(\theta_{wh}) \tag{76a}$$

$$w_{yh} = V_{wh} \cos(\theta_{wh}) \tag{76b}$$

for the estimated wind.

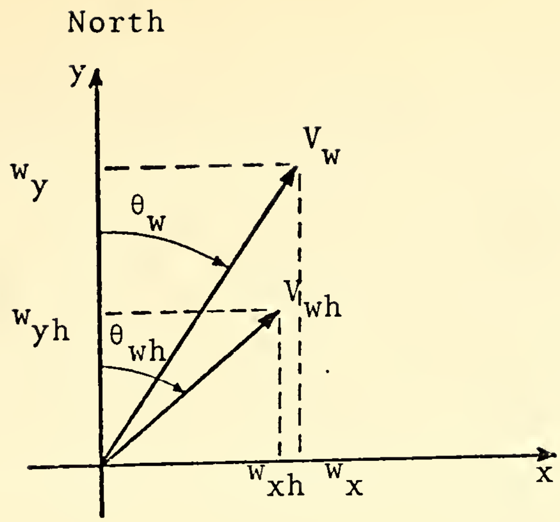


Figure 10. Illustration of Wind Relationships.

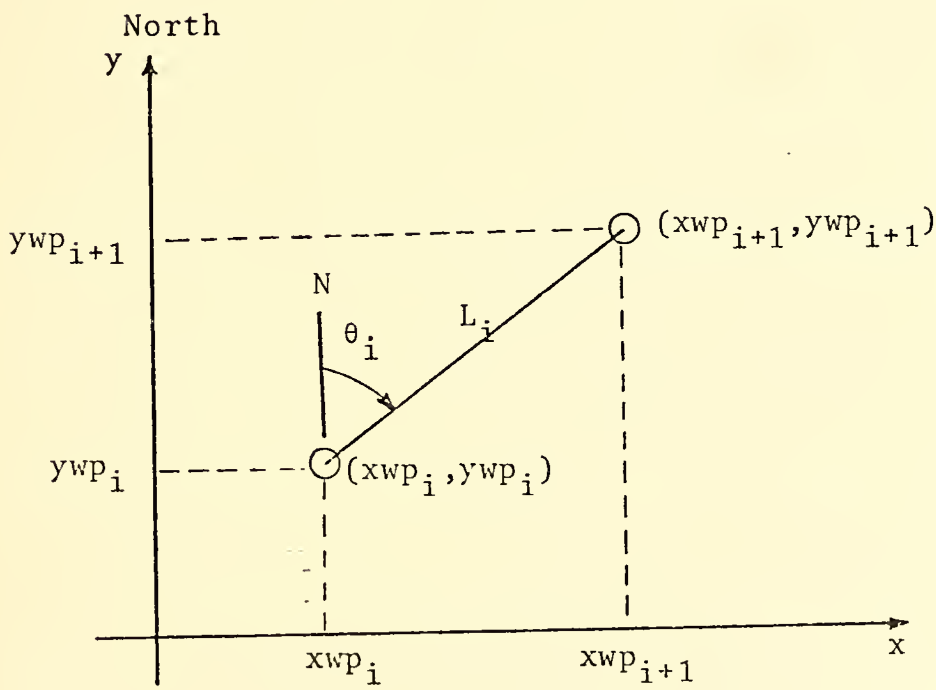


Figure 11. Illustration of Waypoint Coordinate Calculations.

2. Mission Data Table Calculations

Included in the mission data table are the waypoint coordinates, average radar range to each leg, average radar azimuth to a given leg, desired air heading while on each leg, desired ground speed while on each leg, approximate time to fly each leg, and the ground velocity components for each leg.

For notational purposes, it is assumed that there are a total of n legs to be flown; a subscript i on any parameter indicates that parameter for the i^{th} leg. Figure 11 illustrates calculation of the i+1 waypoint coordinates from the previous leg's parameters. If (xwp_i, ywp_i) are the coordinates of the i^{th} waypoint, the azimuth of the i^{th} leg is θ_i, and the length of the i^{th} leg is L_i, then

$$xwp_{i+1} = xwp_i + L_i \sin(\theta_i) \tag{77a}$$

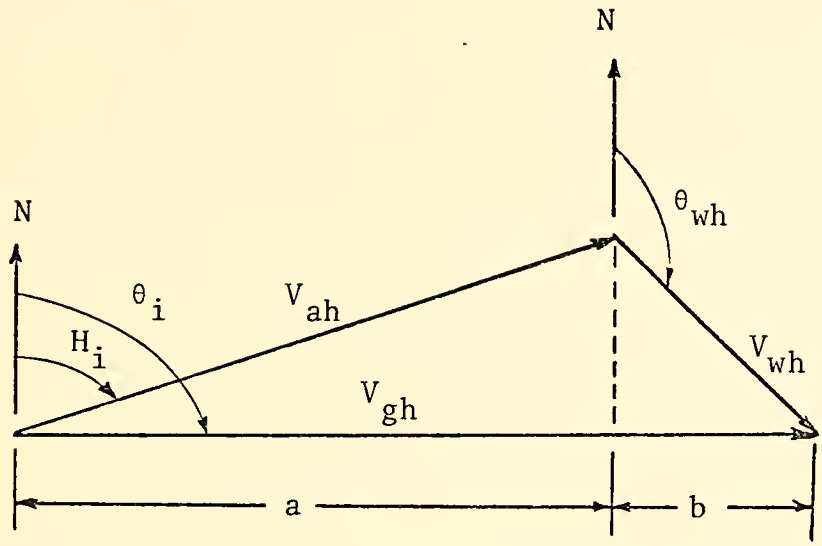
$$ywp_{i+1} = ywp_i + L_i \cos(\theta_i). \tag{77b}$$

The average range and azimuth to the i^{th} leg, R_i and Az_i are given by

$$R_i = \tfrac{1}{2}[(xwp_i+xwp_{i+1})^2 + (ywp_i+ywp_{i+1})^2]^{\frac{1}{2}} \tag{78}$$

$$Az_i = \arctan\left[\frac{xwp_i + xwp_{i+1}}{ywp_i + ywp_{i+1}}\right]. \tag{79}$$

Figure 12 presents geometry which aids in clarifying calculation of the air heading and ground speed, assuming the air speed is a known constant. (This assumption is maintained through the problem and seems reasonable, since



$$a = V_{ah} \cos(H_i - \theta_i)$$

$$b = V_{wh} \cos(\theta_{wh} - \theta_i)$$

Figure 12. Illustration of Ground Speed and Air Heading Calculation Geometry.

the pilot has a direct readout of his speed with respect to the air.) Since it is desired to fly along the leg, it is correct to sum components perpendicular to the leg and set these to zero. If V_{ah} is the estimated air speed and H_i is the air heading then

$$V_{ah}\sin(H_i - \theta_i) + V_{wh}\sin(\theta_{wh} - \theta_i) = 0. \qquad (80)$$

Summing components in the direction of the leg yields the desired ground speed

$$V_{gh} = V_{ah}\cos(H_i - \theta_i) + V_{wh}\cos(\theta_{wh} - \theta_i) \qquad (81)$$

V_{gh} can be broken into Cartesian components, V_{gx} and V_{gy} as follows:

$$V_{gx} = V_{gh}\sin(\theta_i) \qquad (82a)$$

$$V_{gy} = V_{gh}\cos(\theta_i). \qquad (82b)$$

The air heading required to fly in the direction θ_i is found by solving (80) for H_i.

$$H_i = \theta_i - \arcsin\left[\frac{V_{wh}}{V_a}\sin(\theta_{wh} - \theta_i)\right]. \qquad (83)$$

3. Initial Position and Velocity of the Aircraft

The initial true position of the aircraft is that of the first waypoint plus a perturbative error. Since the waypoint is on the ground, the altitude of the aircraft is given by the perturbation in the z coordinate. The true position of the aircraft is contained in the X array.

$$x3 = xwp_1 + \Delta_x \qquad (84a)$$

$$y3 = ywp_1 + \Delta_y \qquad (84b)$$

$$z3 = \Delta_z. \qquad (84c)$$

The initial true ground heading, H_g, is read as data. This in addition to the known airspeed, V_a, specifies the true ground speed, V_g, and the true air heading, H_a, through a set of equations similar to (80), (81), and (83). The results are given by

$$H_a = H_g - \arcsin\left[\frac{V_w}{V_a}\sin(\theta_w - H_g)\right] \qquad (85)$$

and by

$$V_g = V_a \cos(H_a - H_g) + V_w \cos(\theta_w - H_g). \qquad (86)$$

The initial true velocity of the aircraft is then broken into Cartesian coordinates and stored in the XD3 array.

$$\dot{x}3 = V_g \sin(H_g) \qquad (87a)$$

$$\dot{y}3 = V_g \cos(H_g) \qquad (87b)$$

$$\dot{z}3 = 0. \qquad (87c)$$

Note that it is through specification of H_g different from θ_1, and Δ_x and Δ_y different from 0 that nonzero perturbations in velocity and position from the desired values are entered. Figure 9 also shows the geometry which might exist in the case of nonzero displacement and velocity from the desired track.

C. AIRCRAFT POSITION, VELOCITY, AND ERROR ESTIMATION

As previously stated, aircraft motion is simulated by the use of a slightly modified version of subroutine ARCRFT associated with Precision Guidance. Position and velocity estimation is accomplished using RADAR6. At the beginning of

the main processing loop, true aircraft position is updated followed immediately by a prediction update on estimated aircraft position. A new estimation update is performed only if the total radar sampling interval DTRAD has ellapsed. (It is assumed that prediction updates occur at a higher rate than radar sampling.) The most current aircraft position and velocity estimates are contained in the X1 and XD1 arrays. From this and the estimated wind components, the estimated air and ground headings, and estimated ground speed are computed; H_{ah}, H_{gh}, and V_{gh}, respectively.

$$H_{gh} = \arctan\left[\frac{\dot{x}1}{\dot{y}1}\right] \tag{88}$$

$$H_{ah} = \arctan\left[\frac{\dot{x}1 - w_{xh}}{\dot{y}1 - w_{yh}}\right] \tag{89}$$

$$V_{gh} = (\dot{x}1^2 + \dot{y}1^2)^{\frac{1}{2}}. \tag{90}$$

These parameters can be monitored to determine the degree of error in heading and speed from the desired values. An additional and important parameter to be monitored is the extent to which the aircraft has deviated from the desired course, E_{est}. Figure 13 illustrates the geometry used in this calculation. The error is given by

$$E_{est} = (x1 - xwp_i)\cos(\theta_i) - (y1 - ywp_i)\sin(\theta_i). \tag{91}$$

Note that (91) computes the distance from the present position estimate to the leg i. In a command turn, the aircraft should not be on either leg, but will be somewhere between

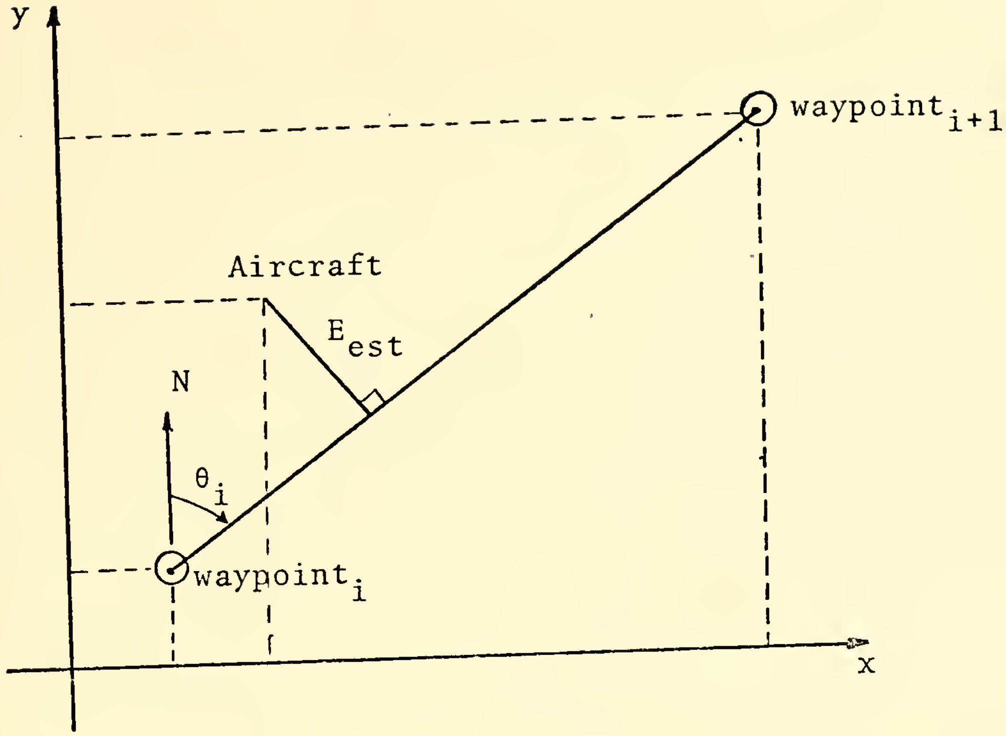


Figure 13. Illustration of Geometry for Calculating Deviation from the Desired Leg Path.

the legs. For this reason, distance from both the i^{th} and i+1 leg is computed, and the smaller of the two values chosen.

True headings, speeds, and distances corresponding to those given above are also computed using the same equations with X3 substituted for X1, and W substituted for W_h.

D. COMMAND TURN CALCULATIONS

The command turn is that which is computed to cause a smooth transition of the aircraft from the present leg to the next leg with a minimum or overshoot or undershoot of the desired path. The aircraft autopilot is constrained to a maximum bank angle, ϕ_m, typically equal to 30 degrees. Since the heading angle rate is proportional to the bank angle in a coordinated turn, the time to complete the turn using the maximum possible bank must be precalculated so that the turn can be started prior to reaching the new leg. This process is complicated by three factors. The first is that the time to the next leg is variable with the current position and velocity. Thus the time to begin the turn as well as the amount of heading change required is a function of the state estimate. The second factor is that the equation to be solving for the amount of time required for the turn is a transendental equation and must be solved through iteration. The third factor is that this equation becomes unduly complex if the turn is not started from a zero bank angle. This requires that the aircraft begin the turn in level flight.

1. <u>Time Remaining on Present Leg</u>

The time remaining on the present leg before intersecting with either the next leg or its extension is computed by finding the intersection of the two paths, calculating the distance to be traversed, and then dividing by the estimated ground speed. The geometry is illustrated in Fig. 14.

The coordinates of intersection of the present path, based on the present velocity estimate and the next leg are designated (x_{int}, y_{int}). Let m_i and m_{i+1} represent the slopes of the present aircraft heading and the next leg, respectively. Then

$$m_i = \dot{y}1/\dot{x}1 \tag{92}$$

$$m_{i+1} = (ywp_{i+1} - ywp_i)/(xwp_{i+1} - xwp_i). \tag{93}$$

The equations of the two lines whose intersection is to be found are

$$y = m_{i+1}\, x - (m_{i+1}\, xwp_{i+1} - ywp_{i+1}) \tag{94}$$

$$y = m_i\, x - (m_i\, x1 - y1). \tag{95}$$

These equations are solved simultaneously to give the point of intersection.

$$x_{int} = \frac{(y1 - ywp_{i+1}) + (m_{i+1}\, xwp_{i+1} - m_i\, x1)}{(m_{i+1} - m_i)} \tag{96}$$

$$y_{int} = m_i\, x_{int} - m_i\, x1 + y1. \tag{97}$$

The distance between x1 and the point of intersection is then

$$D_{tg} = [(x_{int} - x1)^2 + (y_{int} - y1)^2]^{\frac{1}{2}} \tag{98}$$

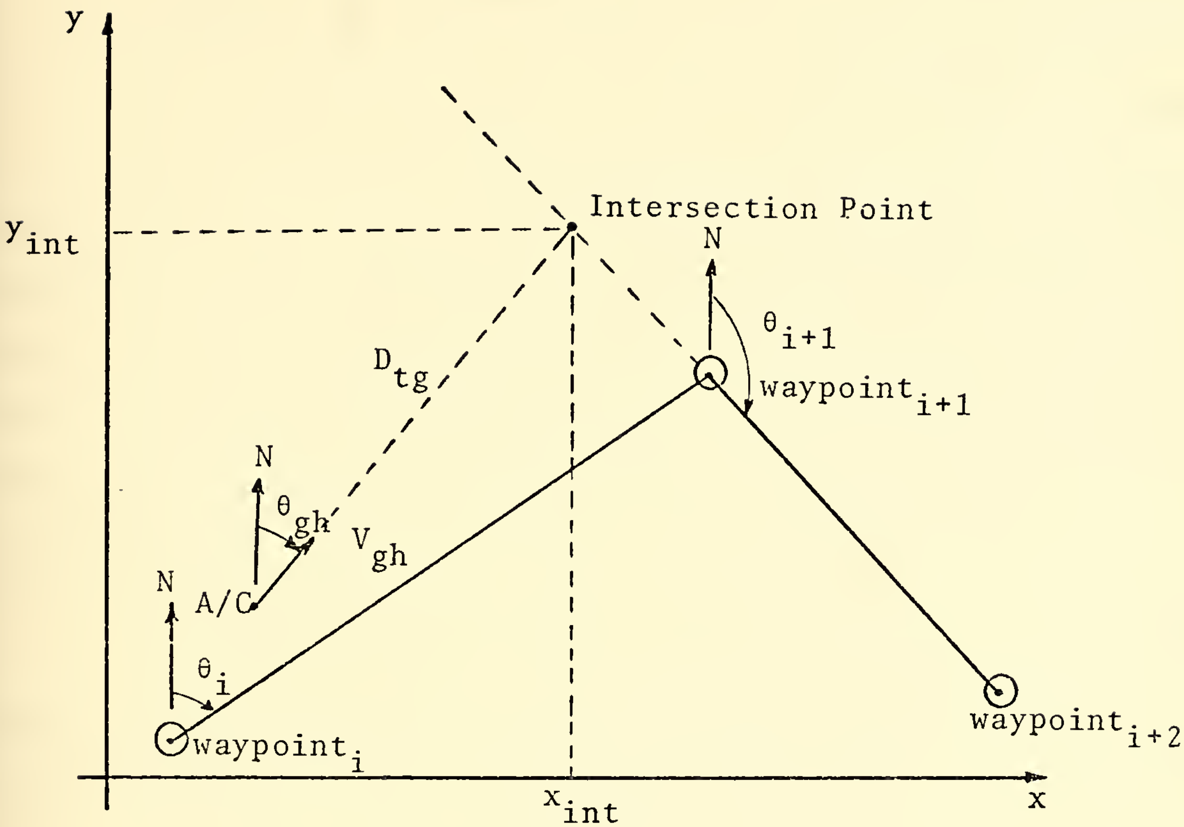


Figure 14. Illustration of Path-Leg Intersection Geometry.

and the time to reach this point is given by

$$TLEG1 = D_{tg}/V_{gh}. \tag{99}$$

2. Time Required to Complete the Turn

The amount of turn required, ΔH, is simply

$$\Delta H = |H_{i+1} - H_{ah}| . \tag{100}$$

Air headings are used in the calculations since all aircraft motion and turn equations must account for the possible presence of wind. Equation (50) gives the relationship between the change in heading angle, time in the turn, and the roll response of the aircraft. Setting $\Delta\psi$ equal to ΔH and ϕ_c equal to ϕ_m gives

$$\Delta H \doteq (g/V_{ah})\left[\phi_m T + (\phi(k-1) - \phi_m)(\tau_b)(1 - e^{-T/\tau_b})\right]. \tag{101}$$

If it is assumed that $\phi(k-1)$ is zero (starting with level flight), then the equation can be rewritten as

$$K \doteq U - (1 - e^{-U}) \tag{102}$$

where

$$U = T/\tau_b \tag{103}$$

and

$$K = \frac{\Delta H \; V_{ah}}{g \; \tau_b \; \phi_m} . \tag{104}$$

For the purpose of finding the turn time required, assume that the turn can be divided into three parts as illustrated in Fig. 15. The first part is the transient period during which the aircraft is coming to the maximum (or minimum) bank angle, going to that limit exponentially with time constant

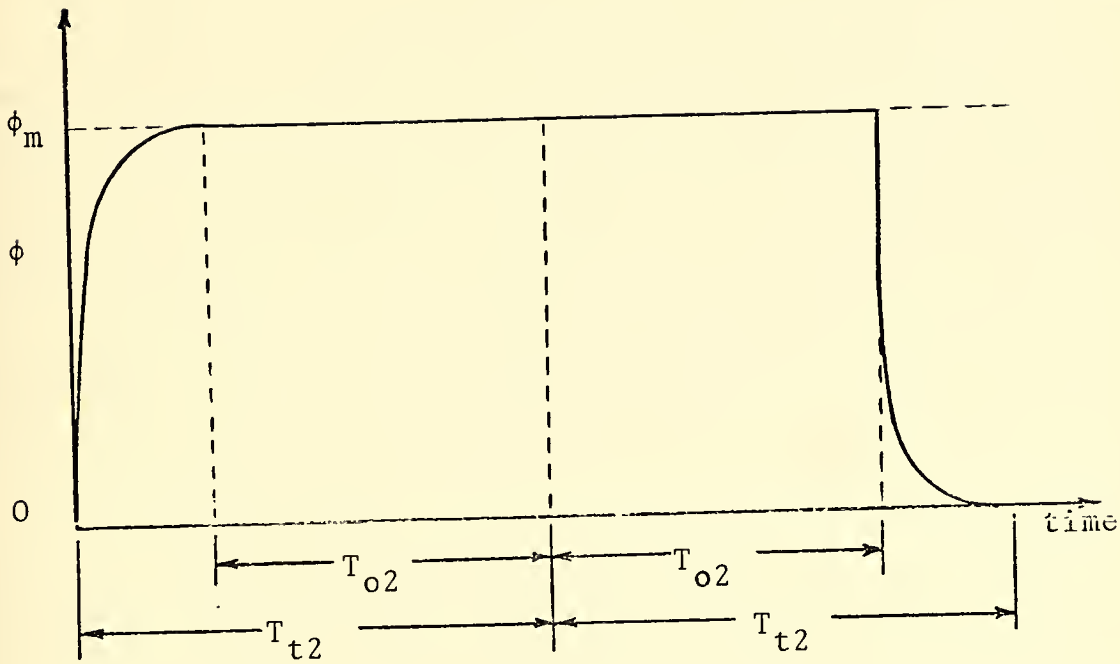


Figure 15. Illustration of Relationships Between Times and Bank Angles in a Command Turn.

τ_b. The second part of the turn is that segment when the aircraft is holding at the bank limit and changing direction with an approximately constant heading rate. The third and final part of the turn is a transient segment during which the aircraft returns to a zero bank angle, also exponentially.

Figure 15 shows that the turn can be divided into two approximately equal parts, during which the aircraft executes approximately half of the turn, $\Delta H/2$. Let the time required to complete half of the turn be T_{t2}, and the time to complete half of the constant bank segment of the turn be T_{o2}. If K, above, is replaced by K/2, then the solution to (102) yields

$$U = T_{t2}/\tau_b. \tag{105}$$

The equation is solved using the Newton-Raphson iterative technique. The initial approximation to T_{t2} is taken from [9]. Iteration continues until the change in T_{t2} is less than 0.01.

Motion of the aircraft through the turn is approximated in nearly the same manner as described in [9]. Briefly, it is assumed that the aircraft continues on an approximately a straight path for a period T_{t2}-T_{02} seconds after the command turn is ordered, followed by the command turn as described above, followed by another period of approximately straight flight along the new leg during the final seconds of the turn. A diagram of the geometry of the turn is presented in Fig. 16. Since the turn has been computed with respect to the motion of the air mass, movement of the air mass during the turn is

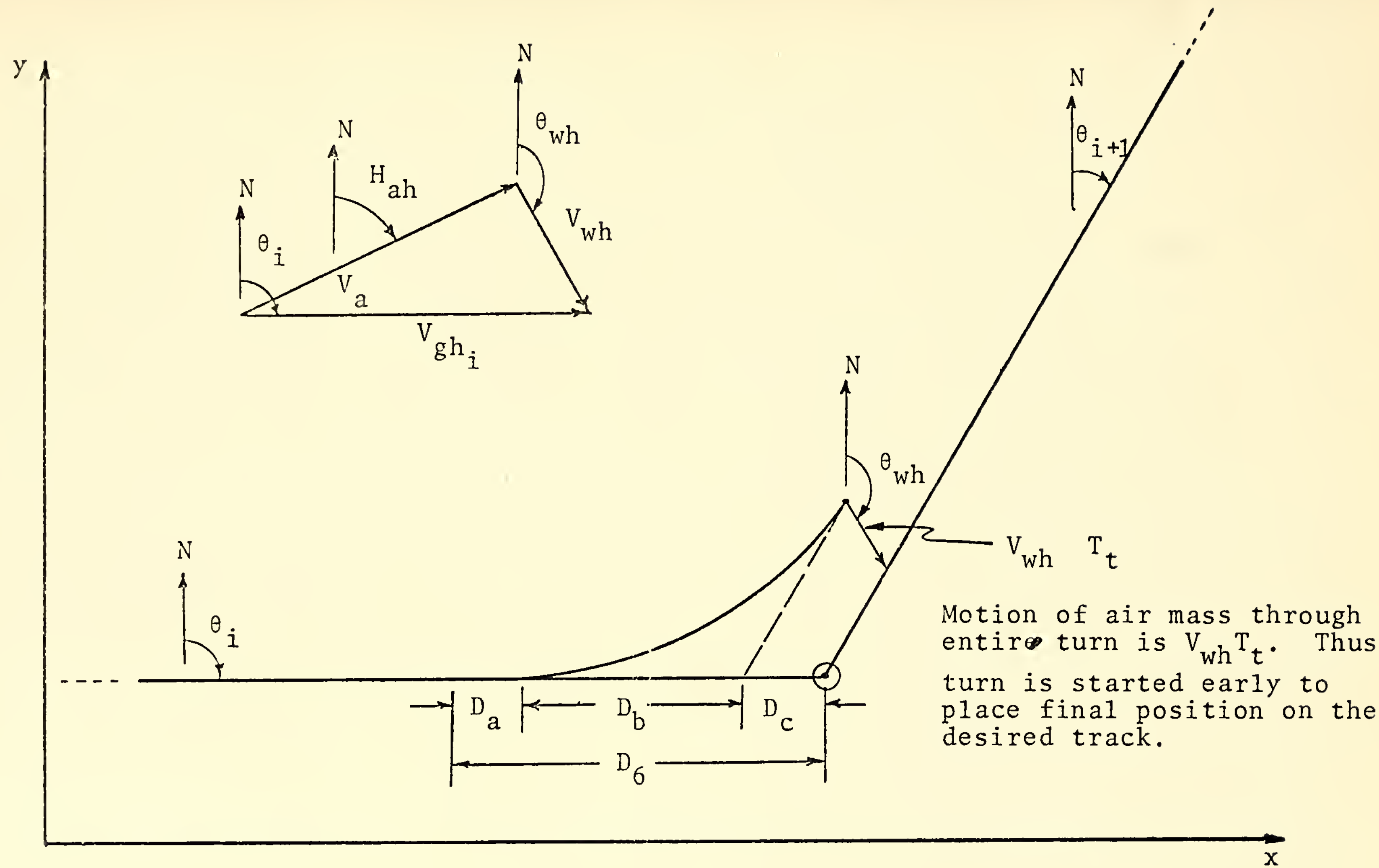


Figure 16. Illustration of Turn Geometry, Including Effects of Wind Shifting the Final Aircraft Position.

compensated for. The only modified equation used to calculate the distance prior to leg intersection from which the turn must start is that which relates to air mass motion. The new equation for computation of the parameter D_c is given correctly by

$$D_c = T_{o2} V_{wh} \sin(\theta_{i+1} - \theta_{wh}) \sin(\theta_{i+1} - \theta_i). \quad (106)$$

The turn must be started a distance D_6 before the intersection with the next leg, where D_6 is shown on the illustration and mathematically defined adequately in [9].

3. Command Turn Timing Logic

The time-to-go before beginning a command turn is given by

$$TG = TLEG1 - D_6/V_{gh}. \quad (107)$$

When TG is less than or equal to zero, the aircraft is commanded to go to the maximum bank angle, with the sign of the bank chosen appropriately. The time to command the bank angle back to zero is T_{stoptn} and is given by

$$T_{stoptn} = T_t - 2\tau_b. \quad (108)$$

Initially it might appear that $3\tau_b$ should be subtracted from the total turn time, since that would be closer to the amount of time required to decrease the full bank angle when the change is occurring exponentially. However, this value was arrived at through simulation trials, and is probably best due to the compounded approximations made in the overall turn solution. The time at which the turn is complete is defined as T_t; no actions depend on this time.

Figure 17 is a logic flow chart of the start/stop turning process. Briefly, a counter in the form of the variable TINTRN is incremented each time through the loop as the turn progresses. The variable ITURN is used as a flag to pass logical control to the correct coding. When no turn is in execution, ITURN = -1, and the only turning which is performed is that to correct the course deviations. No control banking for the above purpose is permitted within the $3\tau_b$ seconds prior to executing a command turn. This is to ensure that the bank angle of the aircraft is zero when beginning the command turn, an assumption which was used in the derivation of the command turn equations. During the command turn, ITURN = 0, and all control logic is bypassed. Immediately upon executing the logic which indicates the turn is ending, ITURN = 1, and calculations to check for the next turn time begin. If required, a new command turn can be executed immediately, before the aircraft has come down from its bank from the previous turn. This feature was added to ensure that if a short leg or a very acute turn was encountered, the best possible flight trajectory would be flown. Examples of the requirement for this feature and its performance are included.

E. AIRCRAFT CONTROLLER DESIGN

The controller, used to guide the aircraft to a desired velocity and keep it there until a command turn, is of the same type used in Precision Guidance. In this case, the controller attempts to keep the estimated ground heading, H_{gh},

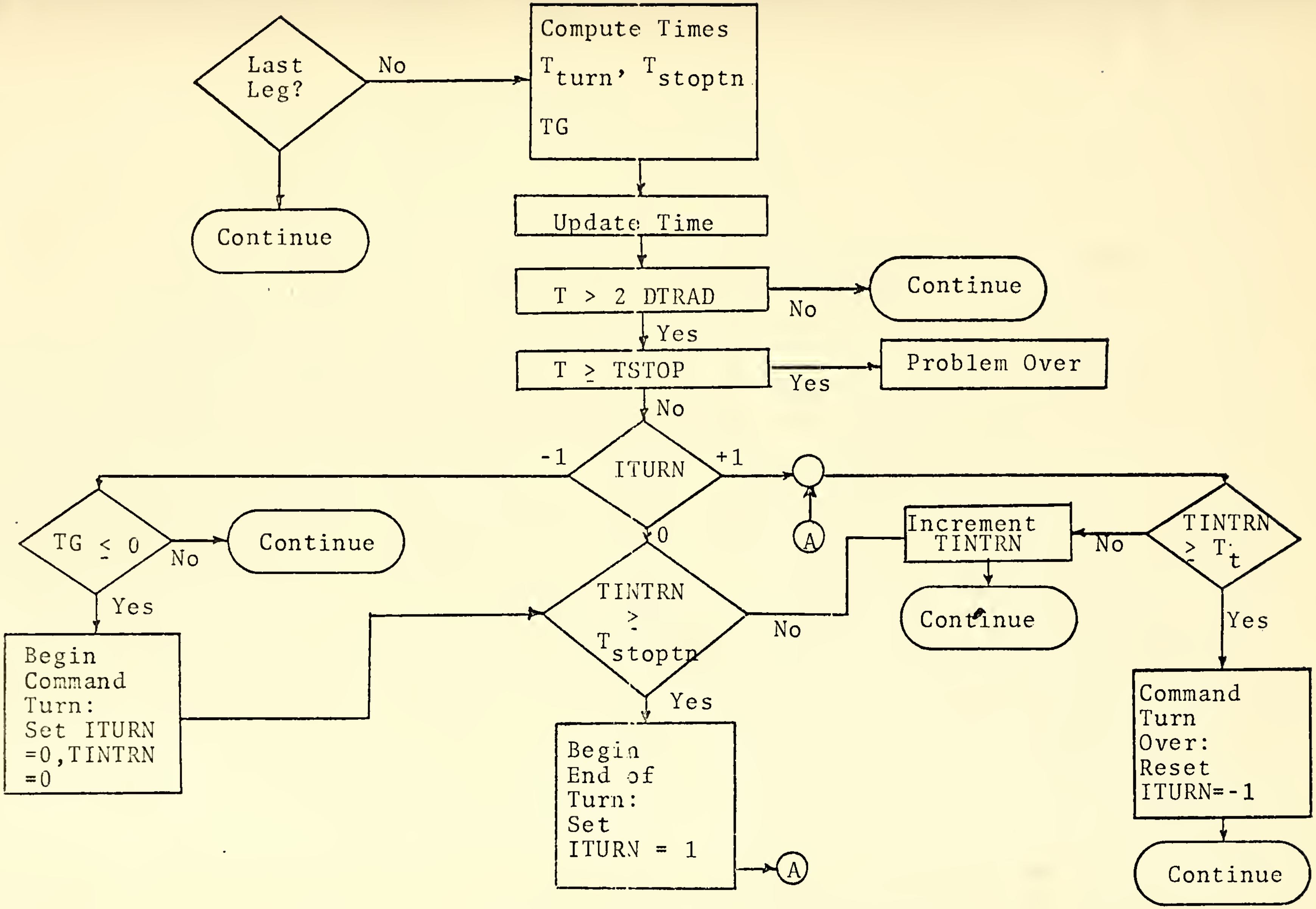


Figure 17. Command Turn Start and Stop Logic.

at that heading from the current estimated position which will fly the aircraft directly at the next waypoint. This scheme was selected for two primary reasons. The first is that it requires minimal control to get the heading correct and thus requires considerably less updating than the technique used in the original simulation program. The second reason is that this technique tends to avoid the problem of oscillation which plagued the original scheme. The controller features a technique which will block any command updates to the aircraft unless the heading error exceeds some minimum error angle, H_{ermin}.

The estimated heading to the next waypoint is

$$H_{hgwpt} = \arctan\left[\frac{xwp_{i+1} - x1}{ywp_{i+1} - y1}\right] . \tag{109}$$

The heading error is then

$$HDE = H_{ghwpt} - H_{hg} \tag{110}$$

and the heading error rate, HDEDOT is the discrete derivative of (110). These are combined as given in (70), (71), and (72) to give the desired control angle. The gains G_1 and G_2 were selected equal, as before, with the value 4. Different feedback gains were used in a number of simulation runs, and the results indicated that the system was somewhat insensitive to the values chosen. The gains yielded desired commands of about 20 degrees during normal course corrections, and with the H_{ermin} feature seemed to avoid the undesired oscillation for the most part.

The commands are quantized before being sent, as in Precision Guidance.

F. COARSE GUIDANCE PROGRAM IMPLEMENTATION

1. Main Program and Subroutines

The Coarse Guidance simulation program consists of a main routine and two primary subroutines. The subroutine ARCRFT used to simulate true aircraft motion, was simplified from that version used in Precision Guidance in that all dive bombing equations were removed. The RADAR6 subroutine differs from RADAR9 by the obvious fact that it does not estimate acceleration due to bank angle bias, as well as in two other ways. The first is that RADAR6 is designed to operate at a sampling interval greater than the control interval. This requires that the subroutine be called, standard prediction equations executed, and then estimation only if the sampling interval has elapsed. The second difference is in the use of NWLD. NWLD indicates that "wild points" are in effect, and are ignored by the radar filter. This amounts to simple prediction without the benefit of radar sampled data.

The Coarse Guidance radar receives little if any return from the aircraft during command turns. In an effort to simulate this effect, NWLD is set to a negative value during turns, thus requiring prediction of position and velocity based only on past data and commands.

Coarse Guidance requires a little less than 100 K of core for linkage and execution. Compilation of the program

and subroutines requires about 25 seconds, and execution requires between 5 and 25 seconds, depending on the number of legs and the leg lengths.

A listing of the variables used in the program along with the variable definitions is provided at the end of Appendix A.

2. Program Input and Output

Specific format definitions for program input data are provided in Appendix C.

The program output consists of three primary parts. The first part is a listing of all input data and simulation run parameters. The second part is a summary of critical parameters printed once per second, and the third part is a summary of the deviation error from the desired path, along with a plot showing the desired, true and estimated paths.

At the beginning of each new leg, a summary of data table information is provided about that leg which can then be compared to what actually occurs. As the run progresses, a notification of the three stages of command turning is presented interspersed with the critical parameter summary.

The only performance criteria selected is the root mean square aircraft deviation from the desired path. The plot of the paths shows the deviations very well. The plot scaling is accomplished to maintain the same scale on both axes for a truer visual indication of relative error. All output values are in nautical miles and feet per second, for position and velocity, respectively.

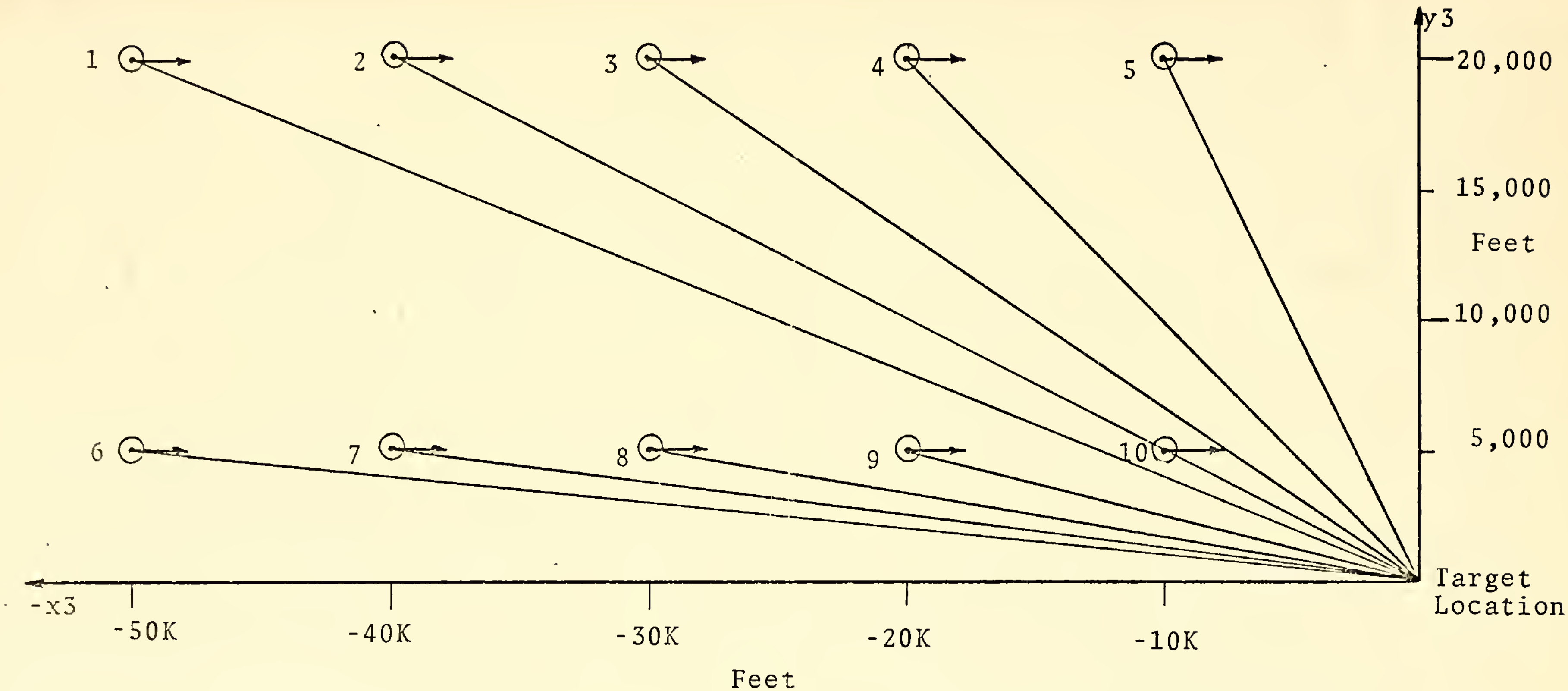
V. PRESENTATION OF RESULTS

A. PRECISION GUIDANCE PERFORMANCE COMPARISON

In the process of improving, changing, and verifying performance of the new version of the Precision Guidance simulation program, a very large number of runs were accomplished. A proper comparison between the two program versions requires side-by-side contrast of appropriate performance parameters, each derived from the same set of initial conditions.

In establishing specific run parameters for the initial conditions, three parameter sets were varied: initial aircraft position, X3, autopilot bias angle, ϕ_b, and aircraft roll response time constant, τ_b. All other variables in the programs were held constant and equal in the two program versions. Variation of the above parameters seemed to yield conditions which would ably show those areas where performance was improved, and at the same time permit runs with initial conditions varying from nearly perfect to considerably in error from the optimal initial bombing path.

Ten different aircraft initial positions were chosen and assigned a "run number." The initial velocity on each of these was identical. The resultant geometry created is shown in Fig. 18. Note that runs 1 through 5 present a considerably more difficult mission than runs 6 through 10. "Difficulty" can be roughly equated to the angle through which the aircraft must change its velocity in order to fly toward the target, located at the origin of the X3 coordinate system.



Initial velocities in each case were the same at 500 ft/sec in the x direction. Note that runs 1 through 5 represent rather extreme initial conditions, and that runs 6 through 10 are more realistic.

Figure 18. Illustration of Initial Position and Velocity Relationships for Runs Included as Results.

1. Filter Performance Comparison

The filters were tested within the framework of the Precision Guidance program. The most important performance parameter to monitor for this problem is the difference between the estimated position of the aircraft and the true position of the aircraft, already defined as the "residual" in II and III. Table I presents a comparison between the Kalman filter and the alpha-beta filter's radial residual, i.e., total estimation error in position, and Fig. 19 presents these results graphically. The numbers shown are average radial residuals, averaged over the runs accomplished for each version.

AVERAGE RADIAL RESIDUAL (ft)

Run No.	Initial x3	Original Version	Improved Version
1.	-50,000	79	15
2.	-40,000	107	17
3.	-30,000	154	20
4.	-20,000	216	21
5.	-10,000	128	27

Table I. Comparison of Average Radial Residuals for the Alpha-Beta and Kalman Filters.

The figures show an average percent improvement for the new program version of 680 percent, with a maximum improvement of over 1000 percent. It should be noted at this point that these numbers do not represent the results of a

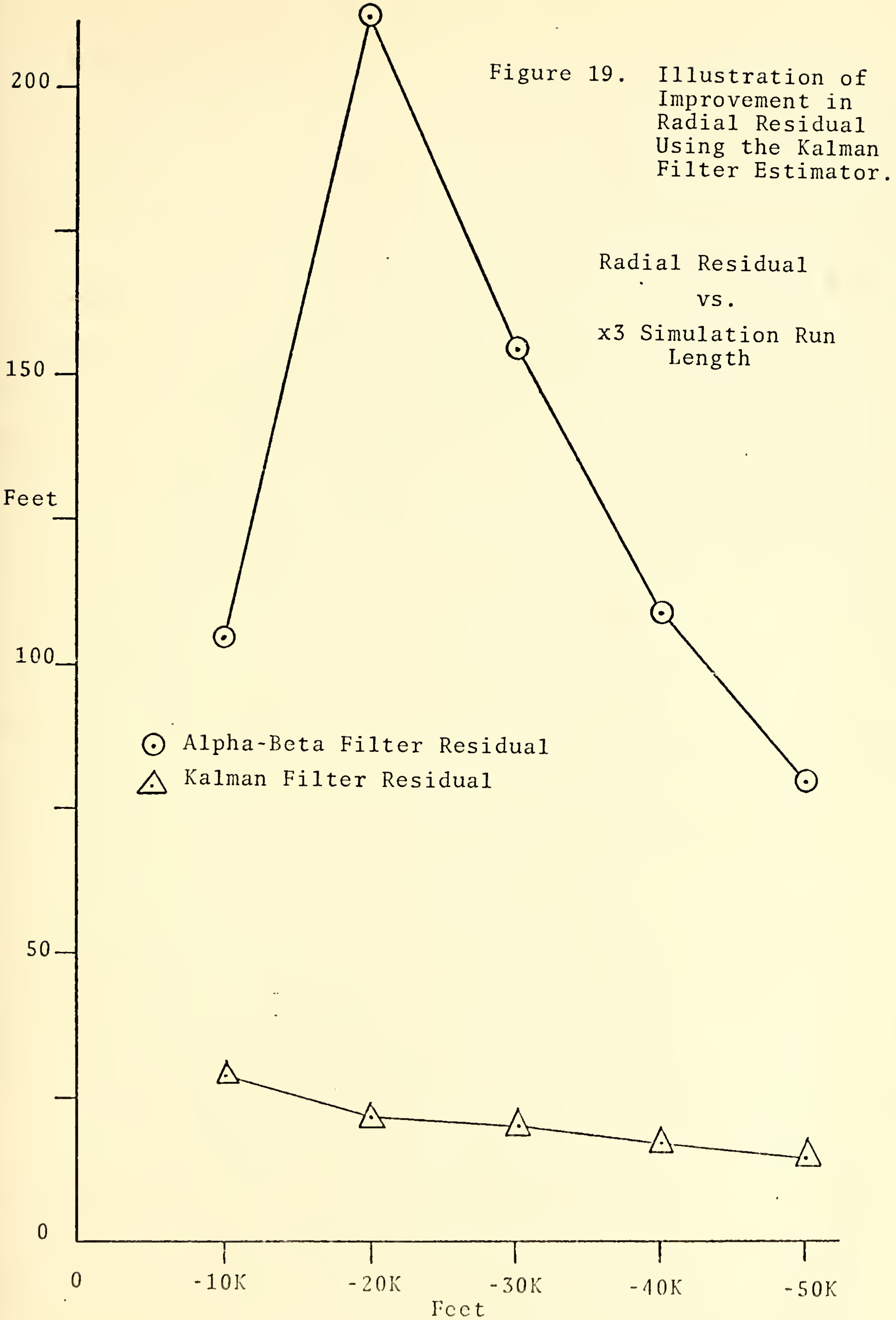


Figure 19. Illustration of Improvement in Radial Residual Using the Kalman Filter Estimator.

true Monte Carlo simulation. To achieve Monte Carlo precision would have required an inordinate amount of computer time to prove or illustrate a point. The differences in accuracy presented in TableIare not simply a result of stochastic luck on obtaining a "good" set of random numbers. The differences in the two versions represents a biasing in the original filter due to the lack of deterministic forcing. This point is brought out even more forcfully in the comparison of Figs. 20 and 21, and Figs. 22 and 23. These plots show that the errors in estimation for the Kalman filter on a typical run are roughly unbiased; however, the biasing on the estimation in both x and y coordinates from the alpha-beta filter is very obvious. Note that, as explained in II, the alpha-beta filter eventually "catches up" to the correct position. This occurs only after the initial period of banking at the limit is complete, and explains the peak in residual error which can be noted on the runs beginning at -20 K ft.

In theory, even if the alpha-beta filter had effects of deterministic forcing included, the Kalman filter would perform in a superior manner, due to its having an "optimal" gain schedule to give a minimum state covariance of error.

2. <u>Bombing Accuracy and Time Response Comparison</u>

Two performance parameters on the overall Precision Guidance routine were observed and minimized throughout program development. The importance of actual bombing accuracy is obvious. A less obvious but very important parameter

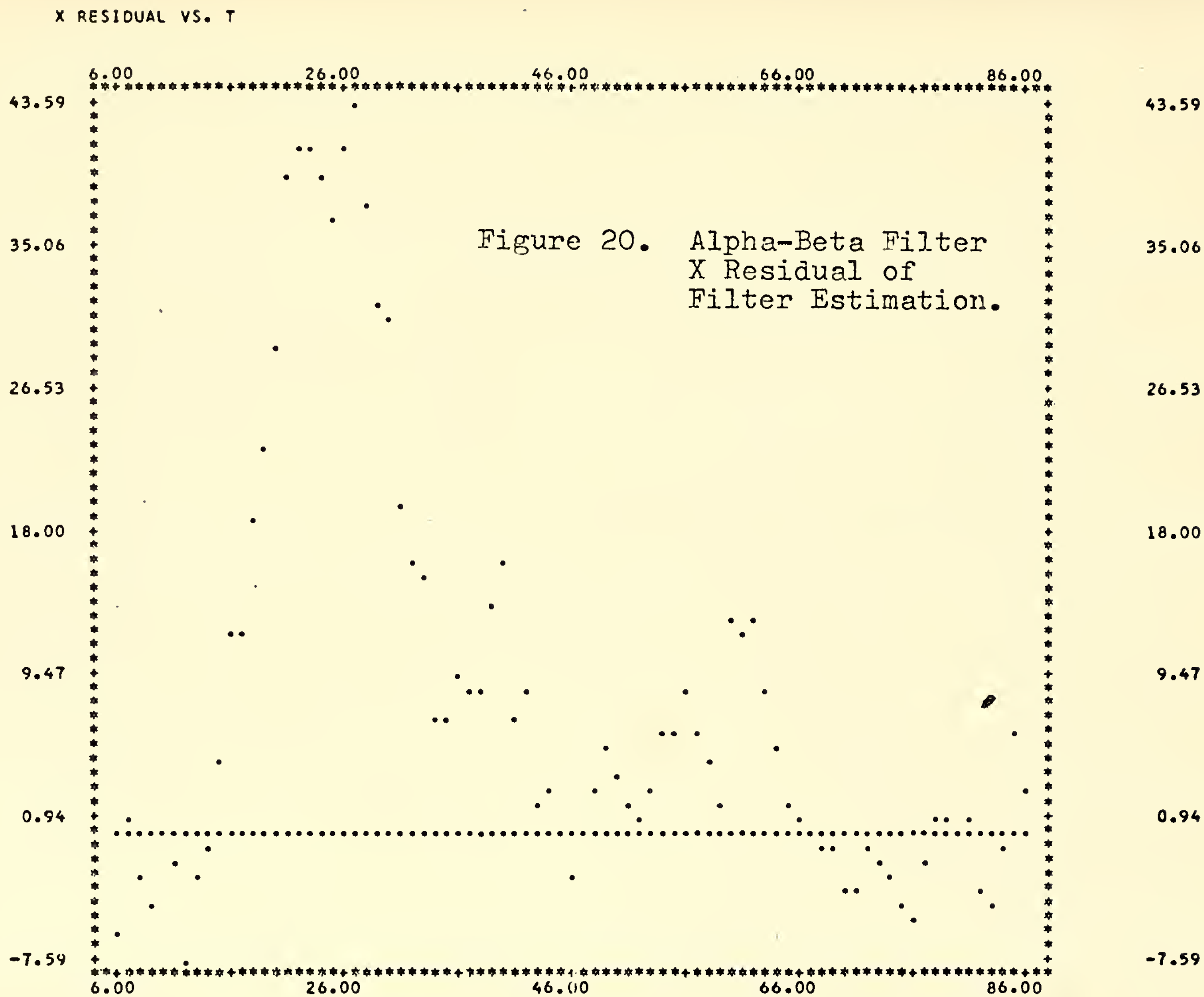


Figure 20. Alpha-Beta Filter X Residual of Filter Estimation.

X RESIDUAL VS. T

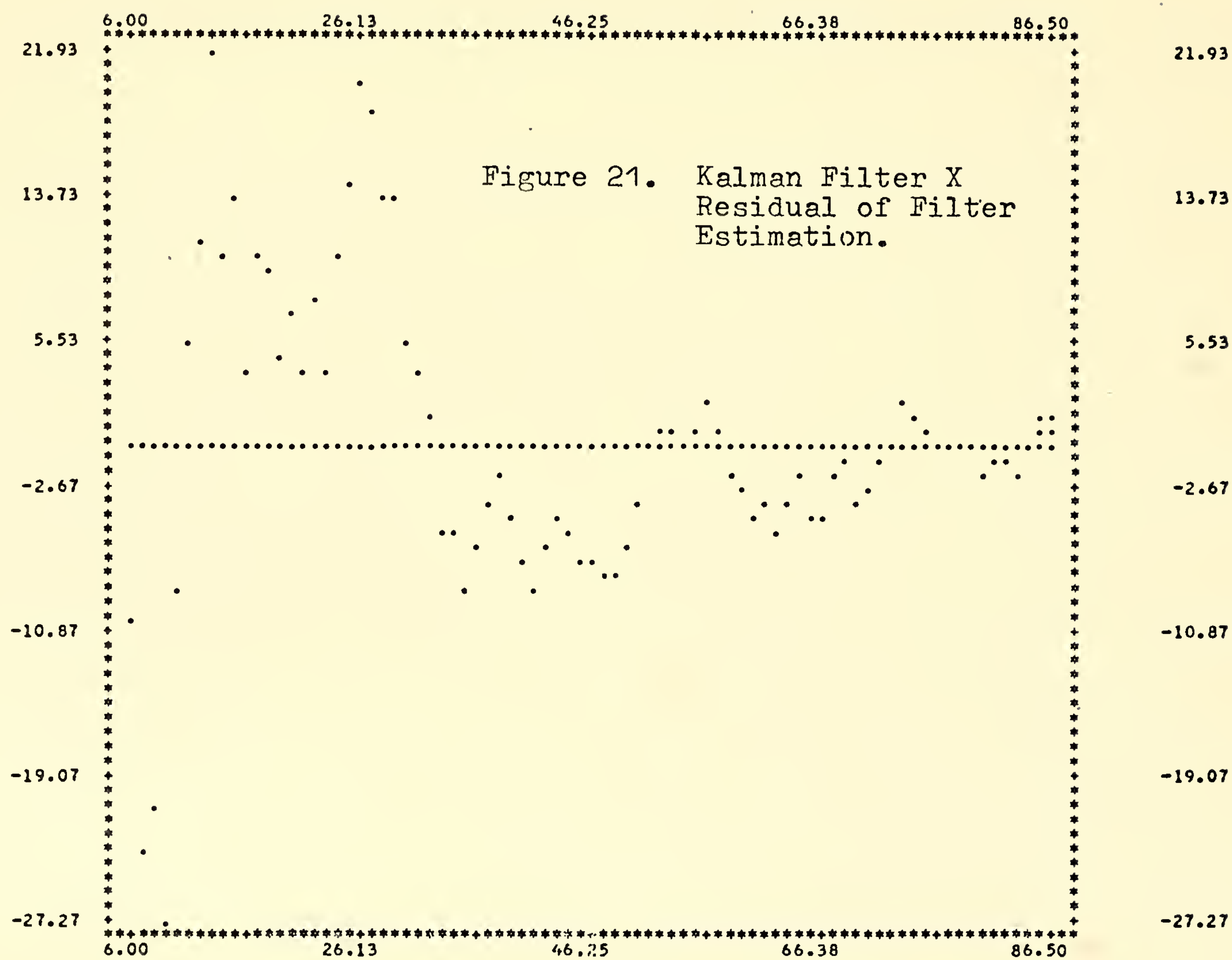


Figure 21. Kalman Filter X Residual of Filter Estimation.

Y RESIDUAL VS. T

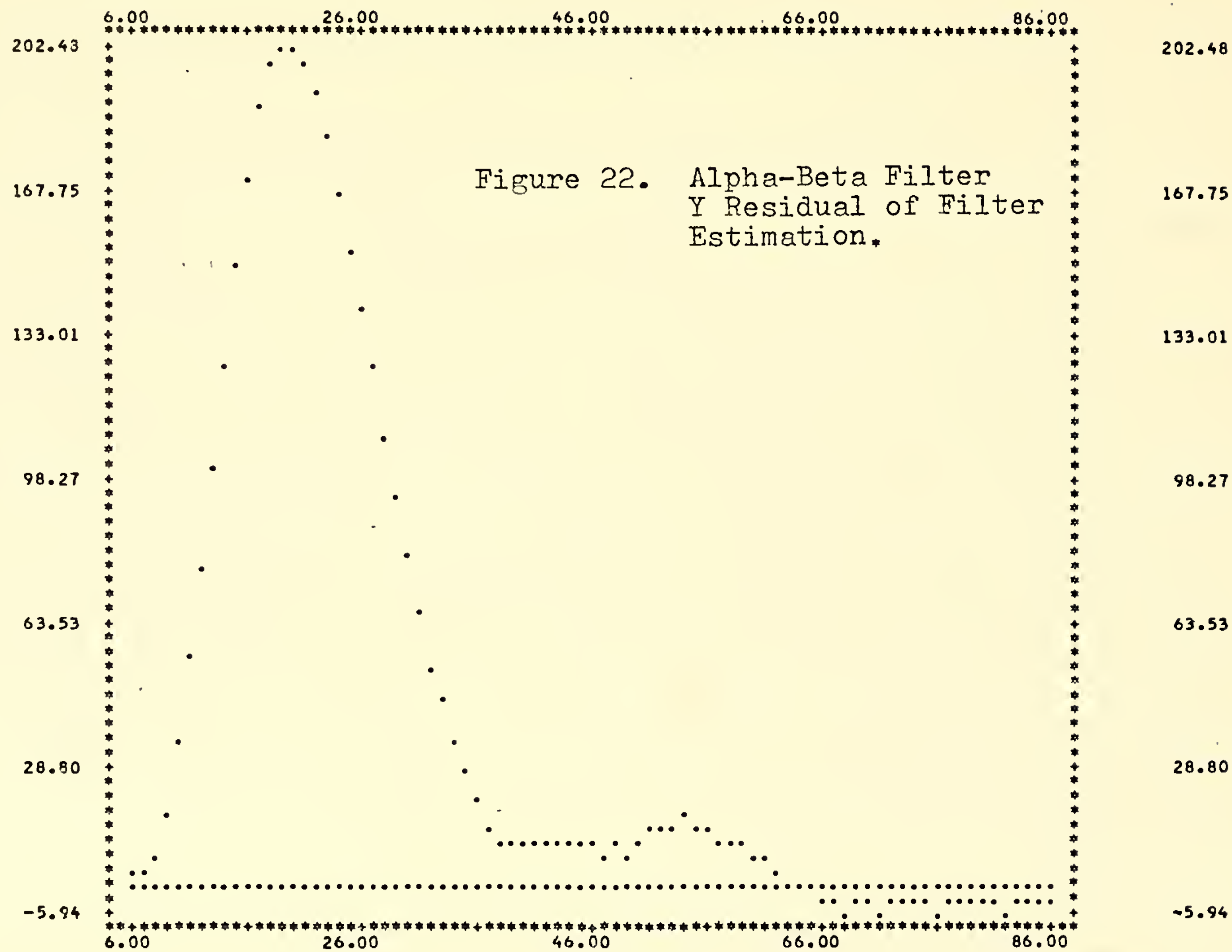


Figure 22. Alpha-Beta Filter Y Residual of Filter Estimation.

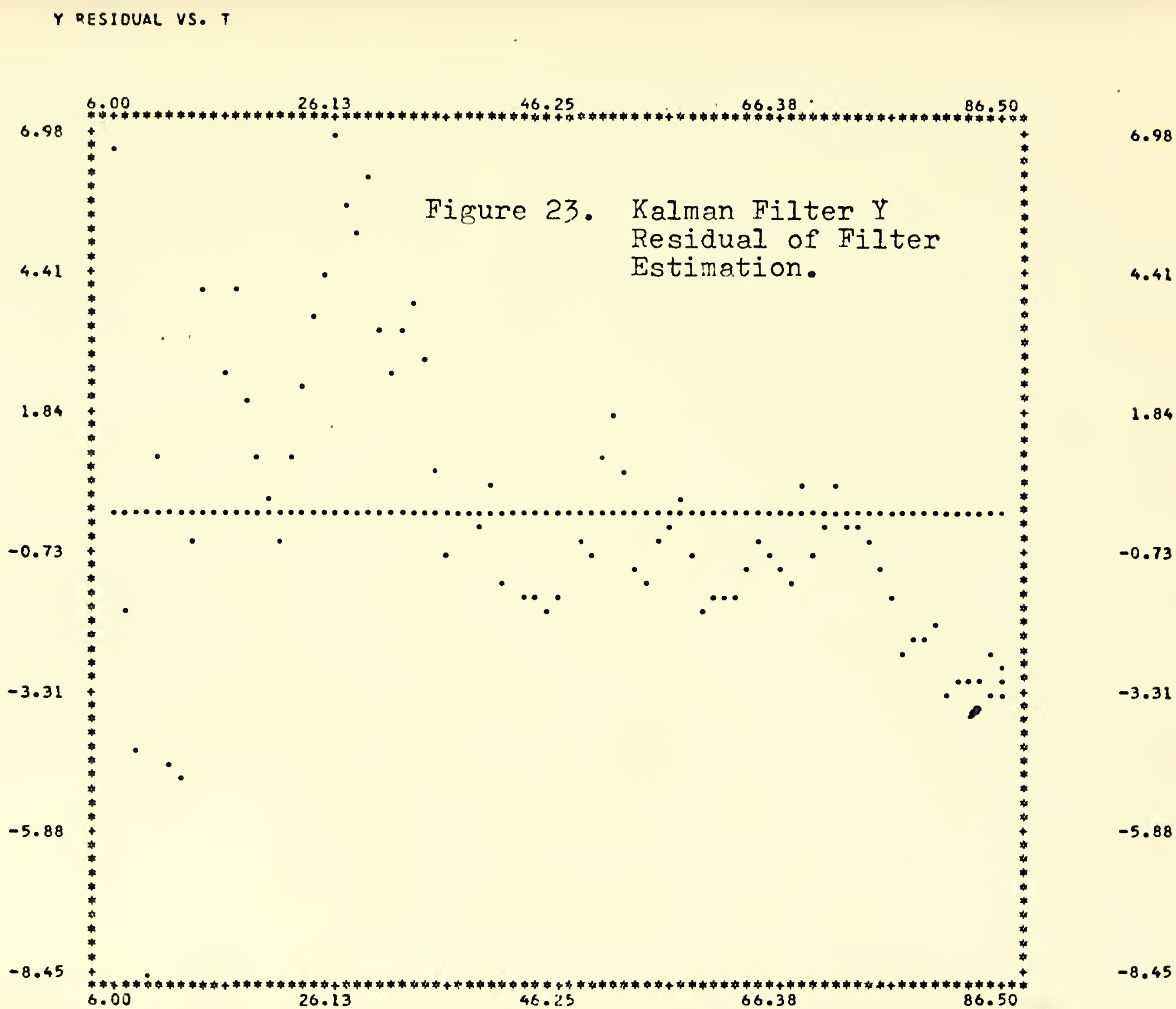


Figure 23. Kalman Filter Y Residual of Filter Estimation.

is the time required on the final bombing leg in order to achieve "sufficient" accuracy. The longer that a pilot is required to fly a straight path toward a target, the more his chances of being shot down increase. It is then desirable to swing onto the final leg from Coarse Guidance and complete the mission as soon as possible.

This time, to be designated T, was difficult to measure directly. It is a strong function of the amount of heading angle change required from Precision Guidance initialization. By varying the angles and distances to the target in the runs (1 through 10), it is possible to observe the amount of time required for the lateral error to reach some threshold minimum acceptable value, under which, in an absolute value sense, the bomb impact will be scored a successful mission. For the purpose of this study, this value was arbitrarily set at 70 feet.

Table II presents a summary of data from the simulation runs illustrating bombing accuracy and the time response parameter T. Note that most of the lists of bombing accuracies have a letter designation at the top of the respective columns. There are referenced on Figures 24 and 25, where comparisons of the bombing accuracies are presented graphically.

The inaccuracies resulting from those runs which start relatively close to the target are due to the fact that the aircraft cannot turn at a high enough rate to get on a bombing line passing near the target before it must drop the bomb. A

Run No. Initial x3	ORIGINAL VERSION				IMPROVED VERSION					
	$\tau_b=2$				$\tau_b=2$				$\tau_b=3.3$	
	$\phi_b=0°$		$\phi_b=3°$		$\phi_b=0°$		$\phi_b=3°$		$\phi_b=3°$	
	T	RI	T	RI	T	RI	T	RI	T	RI
y3 = 20,000 ft		A		B		C		D		
1. -50,000	--	86	--	90	22.5	12	19.5	28	23.5	32
2. -40,000	--	151	--	270	26.5	8	22.5	46	27	55
3. -30,000	--	212	--	331	32.5	9	--	126	--	145
4. -20,000	--	3457	--	1283	--	2208	--	199	--	788
5. -10,000	--	20130	--	20104	--	20394	--	20147	--	20147
y3 = 5,000 ft		E				F				
6. -50,000	68	41			13.5	9				
7. -40,000	--	92			13.5	8				
8. -30,000	--	98			14.5	30				
9. -20,000	--	1120			19	29				
10. -10,000	--	20435			--	6398				

Table II. Precision Guidance Simulation Results. RI is bomb impact miss distance. T is time into run at which the lateral error dropped below 70 ft.

Figure 24. Illustration of Improvement in Bomb Impact Miss Distance for Varying Run Lengths. Initial y3 = 20,000 ft.
Original Program Results: Curves A,B
Improved Program Results: Curves C,D

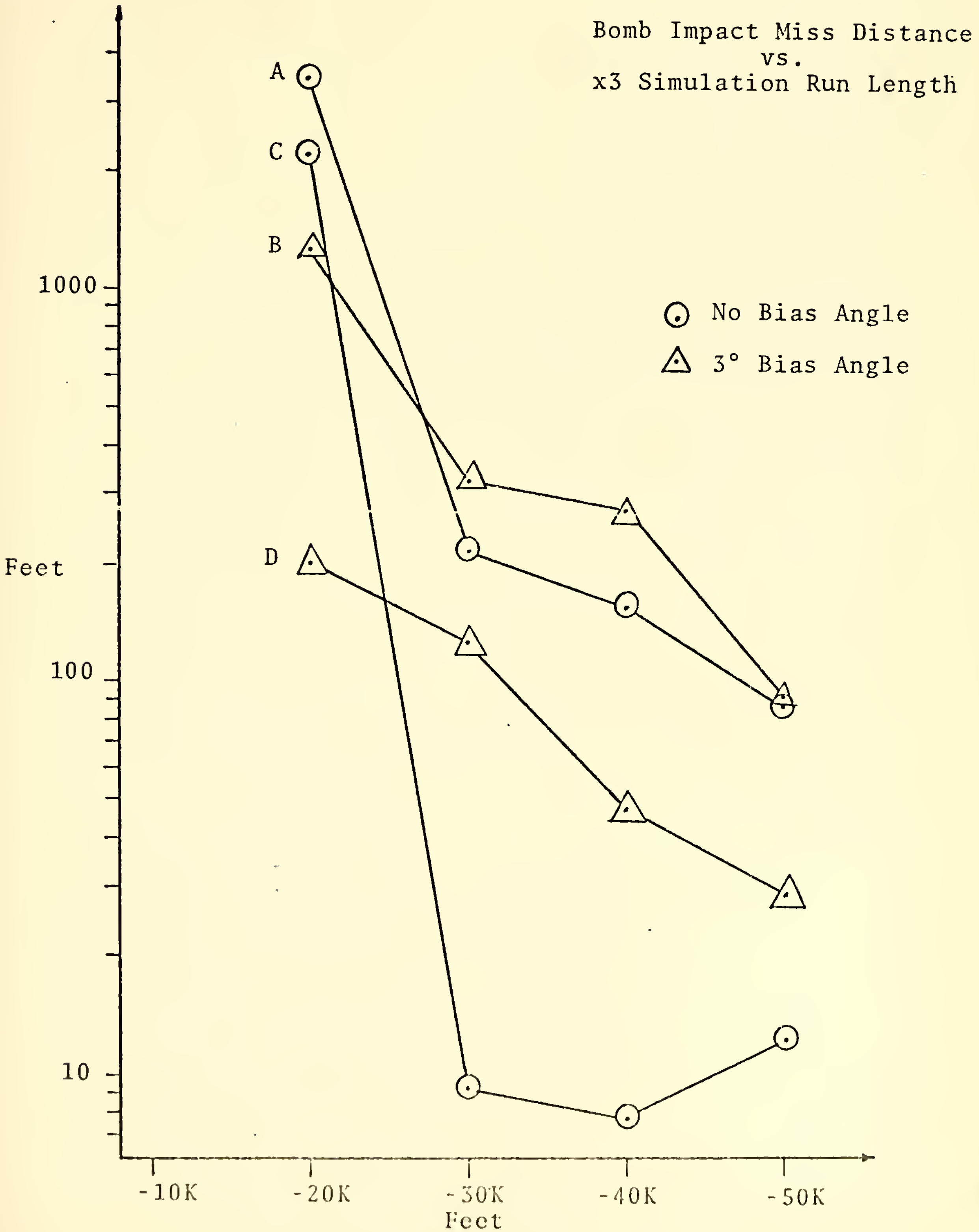
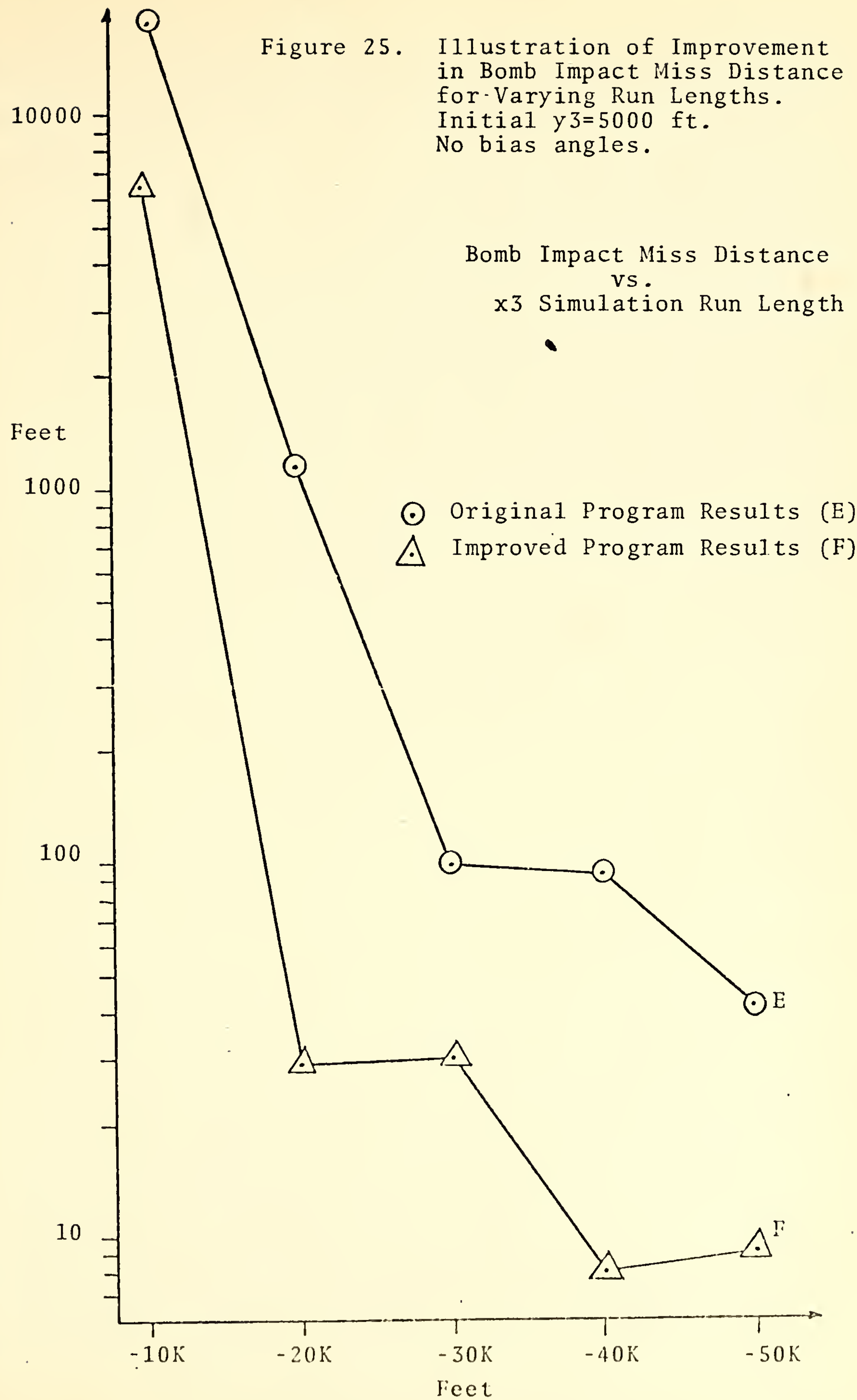
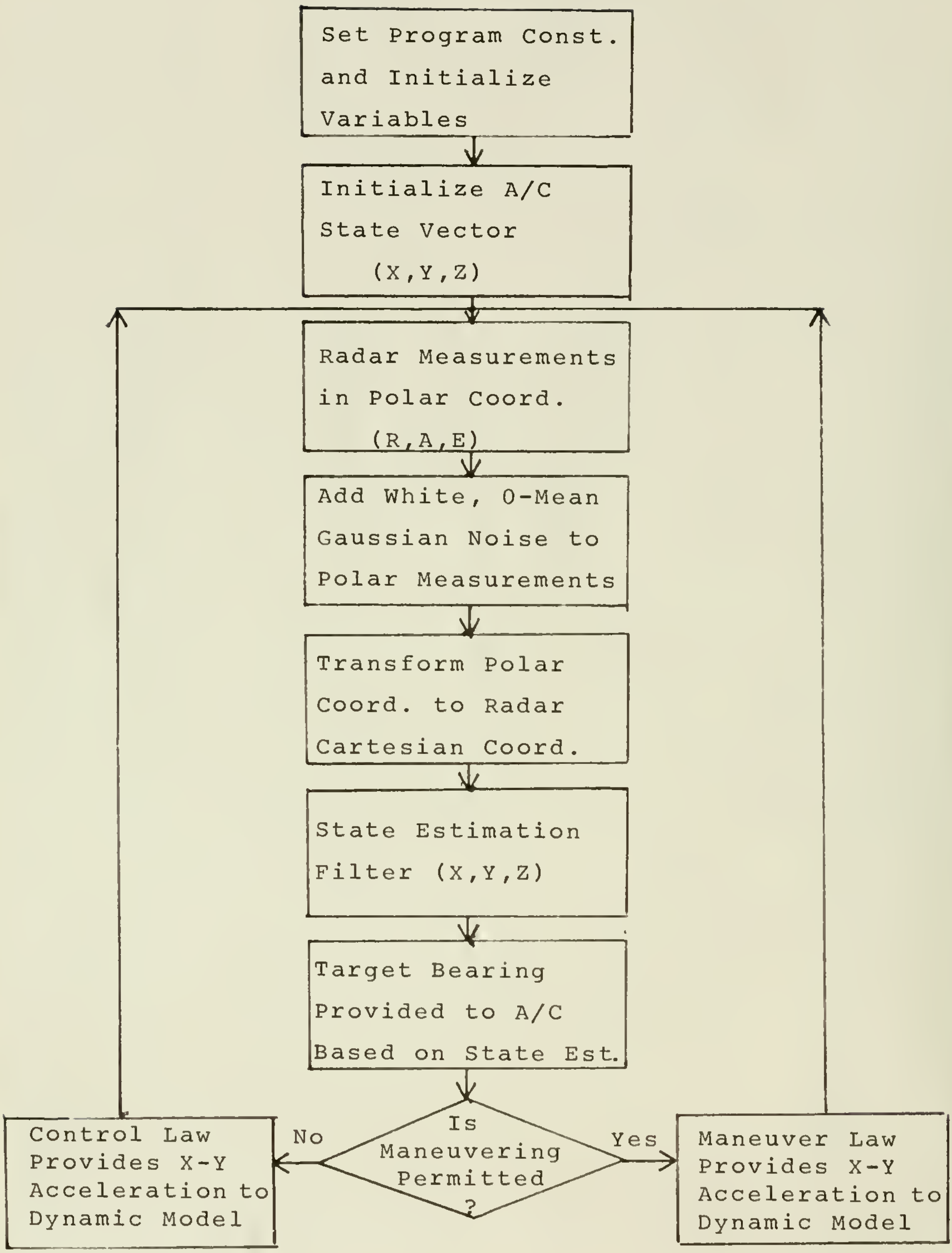


Figure 25. Illustration of Improvement in Bomb Impact Miss Distance for Varying Run Lengths. Initial y3=5000 ft. No bias angles.




Set Program Const. and Initialize Variables
Initialize A/C State Vector (X,Y,Z)
Radar Measurements in Polar Coord. (R,A,E)
Add White, 0-Mean Gaussian Noise to Polar Measurements
Transform Polar Coord. to Radar Cartesian Coord.
State Estimation Filter (X,Y,Z)
Target Bearing Provided to A/C Based on State Est.
Is Maneuvering Permitted ?
No
Yes
Control Law Provides X-Y Acceleration to Dynamic Model
Maneuver Law Provides X-Y Acceleration to Dynamic Model

large part of the error improvement which can be seen in these two figures is due to the filter improvement. However, the controller also has a large effect on the overall response of the aircraft. The use of angle rate feedback helps to drive the lateral error to zero faster than in the original version. This is presented graphically in Fig. 26. Note also that while the original version suffers the effect of a large overshoot past zero lateral error, the improved version does not. This is fundamentally the result of the rate feedback acting as a damper.

Examination of T in Table II whill show that the original simulation program achieved a lateral error below 70 feet only once in the simulation trials run. The improved version reached the acceptable error 9 out of 15 times. Inspection of the times will show that very little time is required to achieve this figure of accuracy, including those runs which are initialized with rather extreme initial error angles (e.g., 3, 8, and 9). Use of the new algorithms to estimate position and control the aircraft should result in a marked improvement in overall accuracy in bombing and in time on the final leg required to achieve this accuracy.

B. COARSE GUIDANCE PERFORMANCE

No contrasting performance tests were performed for the Coarse Guidance mode. The primary reason for this is the lack of sufficient program documentation on the original version of Coarse Guidance. Thus, the Coarse Guidance results

LATERAL ERROR (XE) COMPARISON PLOT

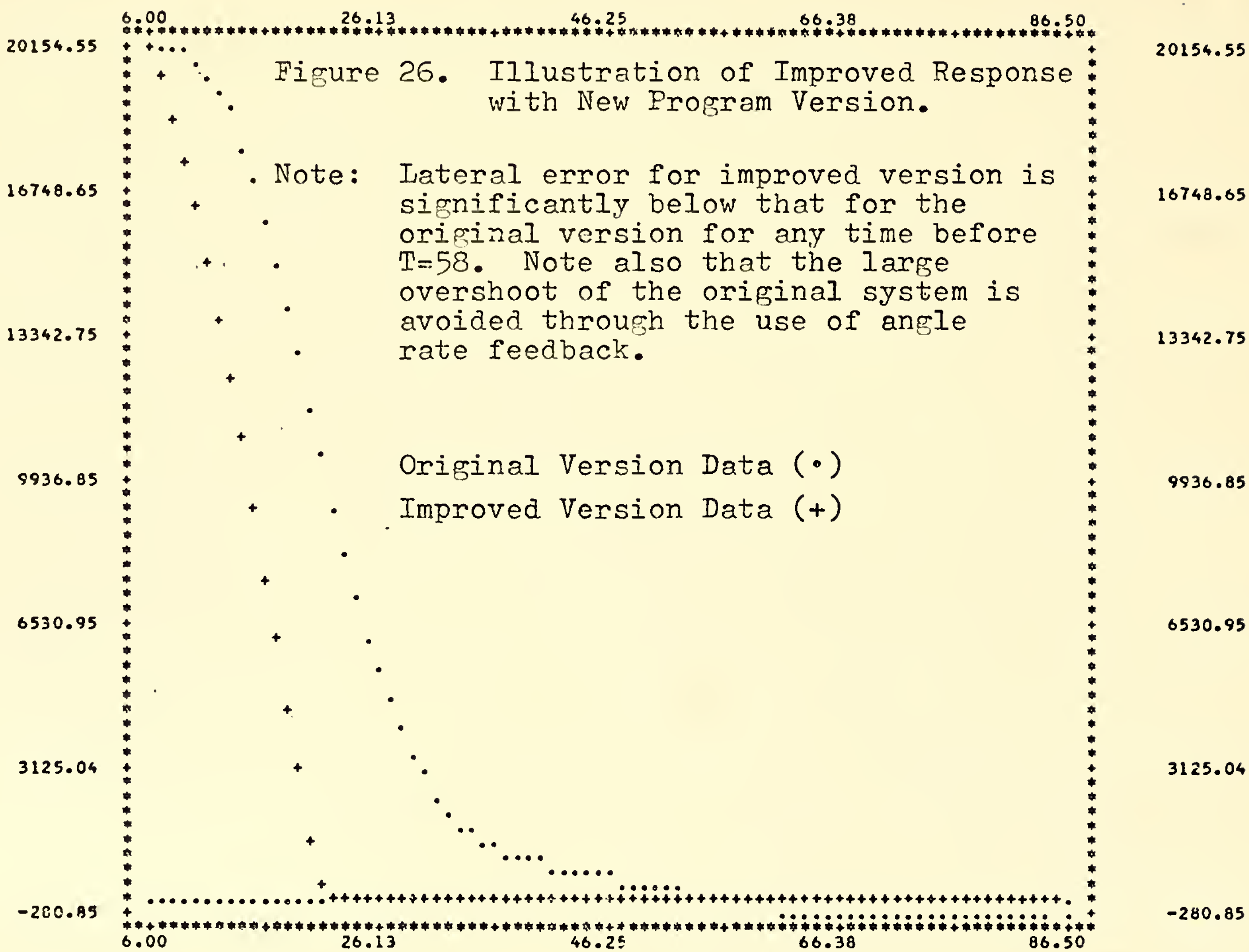


Figure 26. Illustration of Improved Response with New Program Version.

Note: Lateral error for improved version is significantly below that for the original version for any time before T=58. Note also that the large overshoot of the original system is avoided through the use of angle rate feedback.

Original Version Data (•)

Improved Version Data (+)

presented must stand on their own merit, on an absolute instead of a relative performance scale.

As was the case for Precision Guidance, a large number of Coarse Guidance simulation runs were completed in the process of obtaining the final program version. A sample of some of the representative track plots with the RMS track deviations for those runs are presented in Figs. 27 through 30. The significant aspects of each of these runs are discussed below.

Figure 27 represents a rather simple track to follow, and is probably the most realistic track of the four presented. The turns to be executed and the initial heading and position errors are not excessive. The control and turn logic caused the aircraft to follow the desired path very closely.

The scale on Fig. 28 is about half of that for Fig. 27 and thus the initial errors are more obvious. The track deviation figure is higher here than before due to the fact that the initial errors tend to take the aircraft away from the desired track, vice in the track's general direction, as was the case for Fig. 27. Note that the control logic has caused the aircraft to fly toward the second waypoint. At the appropriate time, the aircraft begins a command turn in order to be aligned properly on the second leg upon turn completion. The error from this point on is minimal, never exceeding 0.28 NM on the final leg.

Figure 29 illustrates a most unlikely track to be encountered in practice, but one which checks out the program

DESIRED, TRUE, AND ESTIMATED POSITION

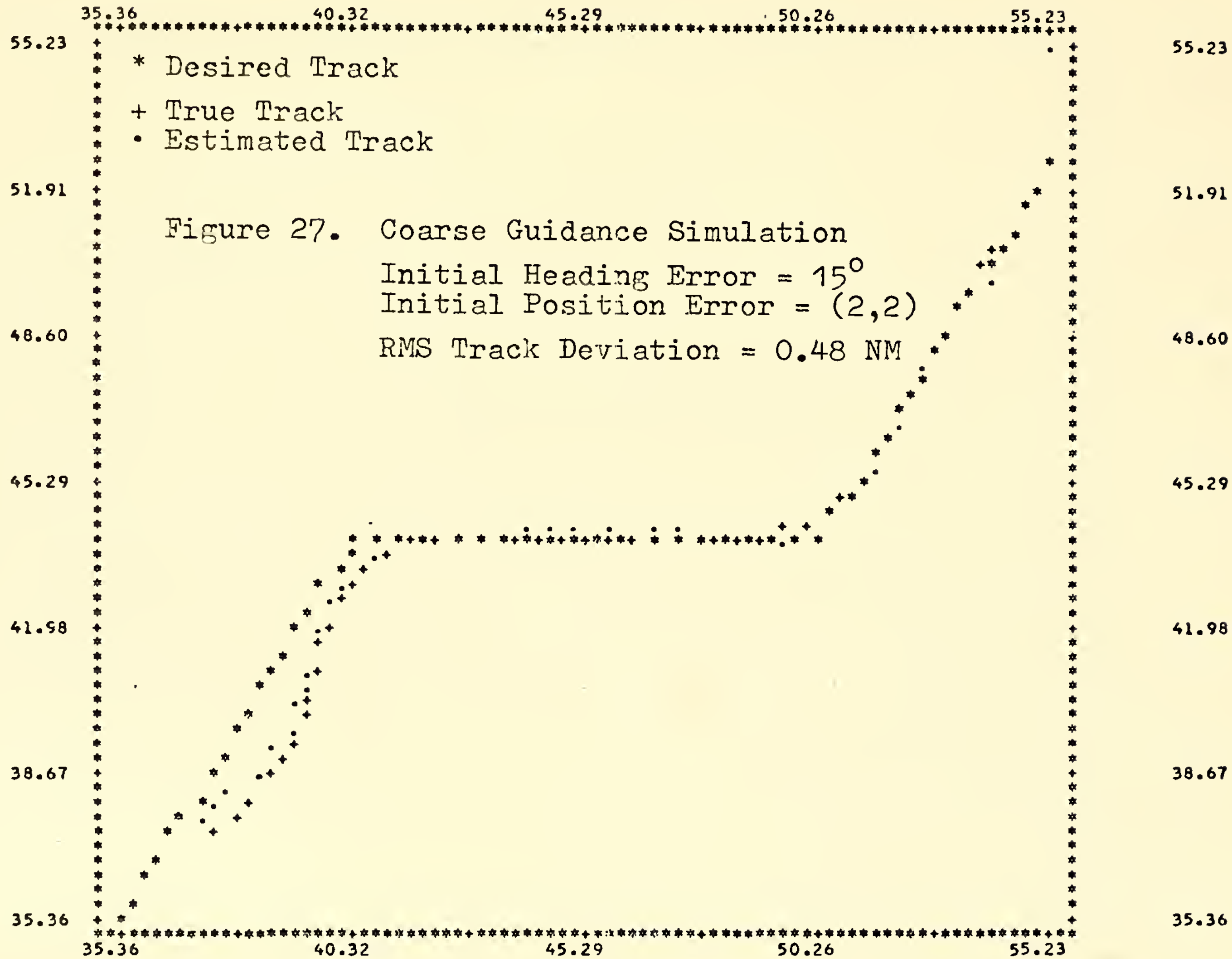


Figure 27. Coarse Guidance Simulation

Initial Heading Error = 15°
Initial Position Error = (2,2)

RMS Track Deviation = 0.48 NM

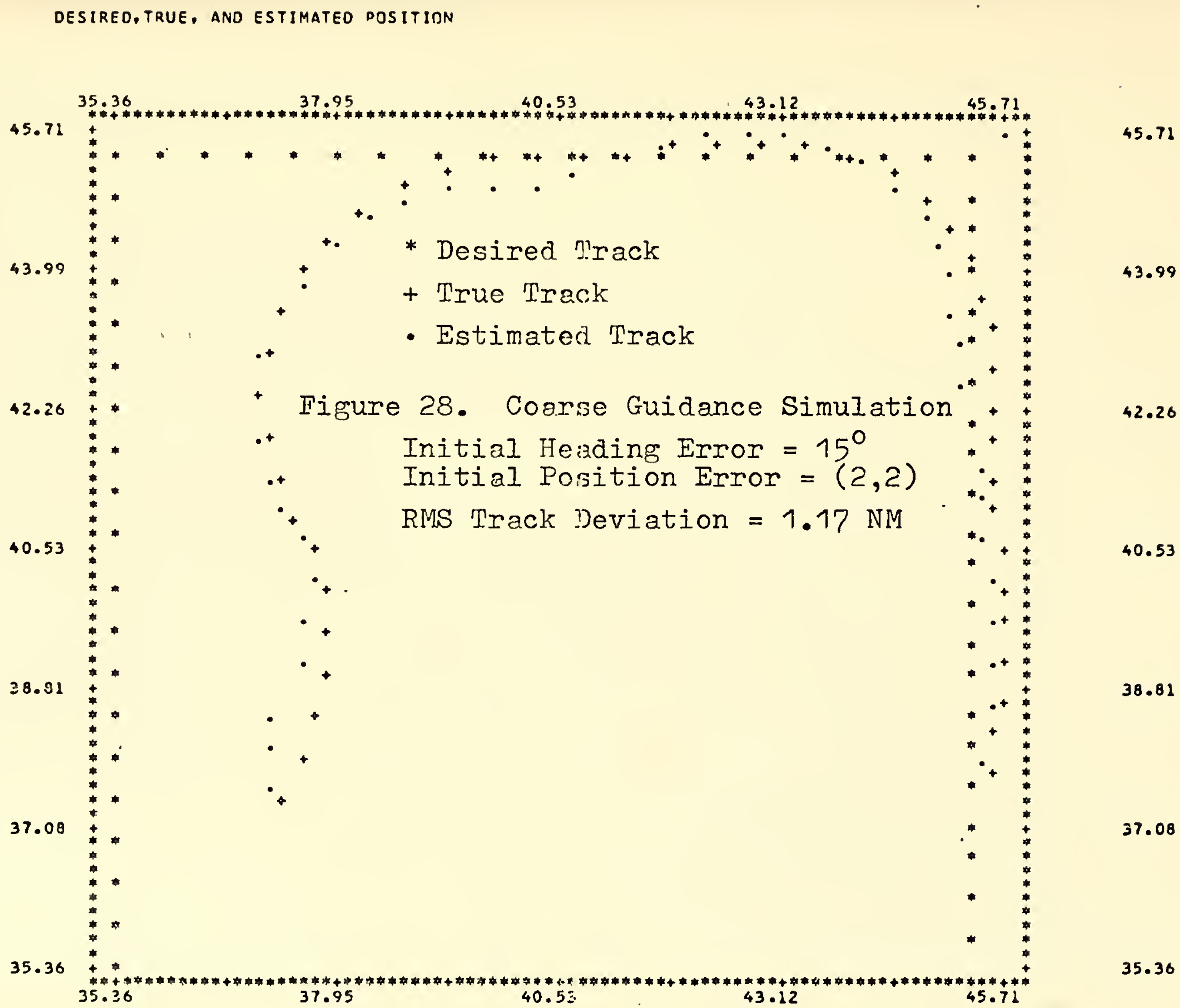


Figure 28. Coarse Guidance Simulation

Initial Heading Error = 15°
Initial Position Error = (2,2)

RMS Track Deviation = 1.17 NM

DESIRED,TRUE, AND ESTIMATED POSITION

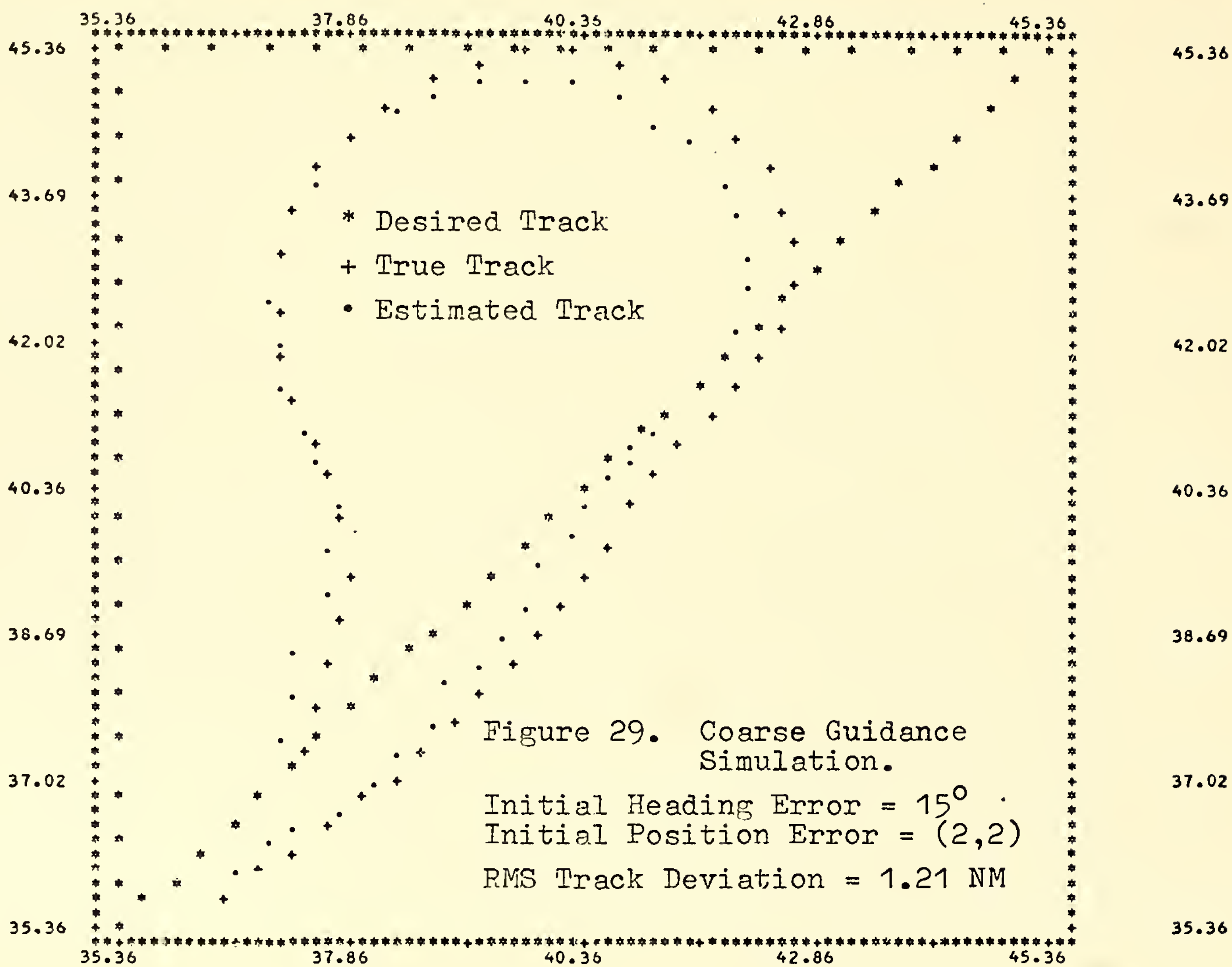


Figure 29. Coarse Guidance Simulation.

Initial Heading Error = 15^{o}
Initial Position Error = (2,2)

RMS Track Deviation = 1.21 NM

algorithms quite well. The initial errors are as before. In this case, the aircraft exists from the first command turn only to note through turning logic that it is already a few seconds behind schedule to be able to make the correct turn onto the final leg. This is because of the combined condition of a relatively short leg length remaining and a requirement to execute 135° change in heading. It can be seen from the plot that the aircraft was not able to come to the correct final leg heading of 225° from the command turn control alone. Another effect which is evident from this run is the lack of position update information in the filter during a command turn, due to poor radar return simulation.

Figure 30 is the result of a track of the same shape as that used in Fig. 29, except the legs are twice as long. The initial position error is also the same as that for Fig. 29, but the initial heading error from the desired track in this case is 90°. The purpose of this track is to show that the program's algorithm can handle effectively a case such as that in Fig. 29, provided the given physical constraints of the system permit time to react.

Note that the aircraft performs the correct course change and heads for the second waypoint, and from that point on is nearly coincident with the desired track. The relatively high track deviation figure for this run is due almost entirely to the effects of the initial heading and position errors.

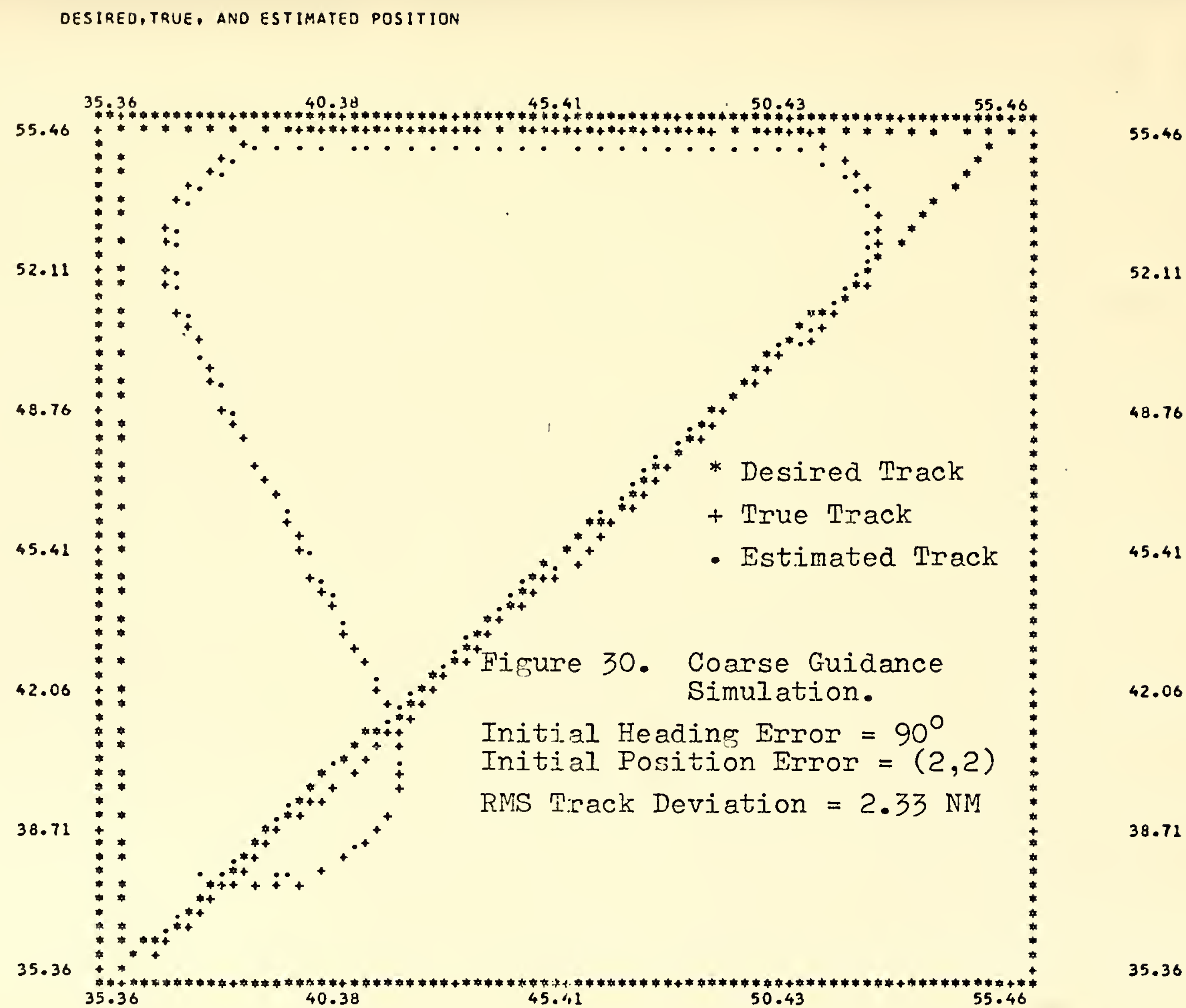


Figure 30. Coarse Guidance Simulation.

Initial Heading Error = 90°
Initial Position Error = (2,2)
RMS Track Deviation = 2.33 NM

Each of these runs was made with a minimum error heading for correction, H_{ermin}, of 5°. Use of this number on runs for which excessive initial errors were not present, i.e., Figs. 27 and 28, resulted in a command correction being sent less than 10 percent of the time during which command turns and initial error corrections were not in progress. This indicates a vast improvement towards the goal of decreasing the numbers of mid-course corrections being sent to the pilot during this phase of the problem.

Autopilot control of the aircraft during this phase was assumed because it was not known how to realistically model the effects of a pilot receiving an instruction to come to a new heading. If pilot control of the aircraft is desired, the algorithms used provide a heading error which could be used. In fact, it is this value which is used to compute the bank angle correction.

In summary, the improved Coarse Guidance algorithms appear to yield a greatly simplified and accurate technique for guiding the aircraft onto the final leg. In addition, the errors in position and heading as the aircraft enters the final leg are such that the Precision Guidance filter can be initialized with valid nonzero velocities, and thus the time required on the final leg can be reduced even further, since the requirement for a full six seconds without control could be reduced.

APPENDIX A: LIST OF PRINCIPLE VARIABLES FOR THE PRECISION AND COARSE GUIDANCE SIMULATION PROGRAMS

A. PRECISION GUIDANCE MAIN ROUTINE

ATITLE-ZTITLE	Titles of the plots
BIS	Means of RADAR9 R, θ, and ϕ noise
BFF	Bomb ballistics form factor
D	Bomb diameter
DEG	Radian/degree conversion factor
DT	Radar sampling interval
DTCON	Control update interval
EM1	Matrix to convert from target to radar reference frame
EM2	Matrix to convert from radar to target frame
EM3	Matrix to rotate from X5 to X6 reference frame
EV1	Translation matrix to go from target to radar reference frame
FILRES	Contains estimation error in radar frame for x,y,z, and radial
G	Gravitational constant, 32.2 ft/sec^2
HDE	Heading error angle
HDEDOT	Time derivative of HDE
HA	Height of target above sea level
IB1	Equal to 0 in level bombing mode
IDTCON	Integer relating DT to DTCON
ITAB1	Counter for number of points to plot
ITH	Equal to the number of times through main processing loop, except on last time through, when equal to -1
IU	Seed for random number generator, NORMAL
KTH	Counter to indicate time for integration for RA and TF calculations
MWLD	Number of "wild points" to be thrown out by RADAR9 as invalid

NWLD	Set equal to MWLD at time TWLD and decremented as "wild points" are used
NTB	Number of bombing constant table to be used
PH	True bank angle
PH1	Estimated bank angle
PHD	Desired feedback command
PHD1	Desired feedback command from HDE
PHD2	Desired feedback command from HDEDOT
PHDAVG	Command to be ordered for last two seconds before bomb release
PHB	Autopilot bias bank angle
PHC	Command bank angle sent
PS	True aircraft heading angle in target frame
PS1	Estimated aircraft heading angle in radar frame
PSD	True aircraft turning rate
RE	Radius of the Earth
RBT	Distance from aircraft (estimated) to the target
SIG(1) - SIG(3)	Standard deviations of R, θ, and ϕ
SIG(4) - SIG(6)	Initial radar velocity estimates
SIGW	Standard deviations of random forcing to be assumed for calculation of Q
T	Time into the simulation run
TB	τ_b, the aircraft roll response time constant
TG	Time-to-Go to release bomb
TF	Time of fall for the bomb
TLVL	Time for required level flight before bomb release; equal to two seconds
TLVL1	Equal to TLVL +1
VE	Total airspeed
VEH	Horizontal airspeed
W	Weight of bomb in pounds
WH	Estimated wind in target frame
WR	Estimated wind in radar frame
WT	True wind in target frame

X1,XD1,XDD1	Estimated position, velocity and acceleration in radar frame
X2,XD2	True position and velocity in radar frame
X3,XD3	True position and velocity in target frame
X5,XD5	Estimated position and relative velocity of target with respect to the aircraft; "primed system"
X6,XD6	X5,XD5 system rotated to align the YD6 axis pointing at the target; for printing and plotting purposes, this vector contains the error between true and estimated position in the X3/X5 frame
XE	Lateral error
XGC	X bomb impact point in X6 frame
XXA-XXG	Arrays used to store variables for plotting at program end
YYA-YYU	Arrays used to store variables for plotting at program end
YGC	Y bomb impact point in the X6 frame

B. SUBROUTINE ARCRFT

CA1 - CA5	Constants used in aircraft motion equations
DT	Radar sampling interval; also interval of update for aircraft true position
DEG	Radian/degree conversion factor
DT3	Equal to DT/2
G	Gravitational acceleration, 32 ft/sec^2
ITH	Equal to number times through loop, except on last time when ITH = -1
IB1	Equal to zero for level flight mode
PH	True bank angle
PHB	Bank angle bias
PHC	Command bank angle
PHN	New bank angle
PS	Heading angle
PSD	Heading angle rate
PSDN	New heading angle rate

PSN	New heading angle
T	Elapsed time since start of run
TB	τ_b, the roll response time constant
TG	Time to go to release bomb
VT	Horizontal airspeed
WT	True wind vector
X3	Aircraft position in target frame
XD3	Aircraft velocity in target frame

C. SUBROUTINES RADAR6 AND RADAR9

A	Azimuth angle
ADDSUB	Subroutine entry to add or subtract two matrices
ADUM	Temporary array used in matrix arithmetic
ANGMAX	Maximum angle, used to prevent overflow
ANGMIN	Minimum angle, used to prevent underflow
BIS	Range, azimuth, and elevation noise bias
BDUM	Temporary array used for matrix arithmetic
CA1,CA4,CA5 CAA2	Constants used in aircraft motion equations
DEG	Radian/degree conversion constant
DT	Radar sampling interval in RADAR9
DTRAD	Radar sampling interval in RADAR6
DTCUM	Accumulator to determine next time to sample position in RADAR6
DT2	Equal to $DT^2/2$
DTRAD2	Equal to $DTRAD^2/2$
DT3	Equal to DT/2
DT4	Equal to $DT^3/6$
E	Elevation angle
E1,E2,E3	Difference between predicted position and data sample during state estimation
G	Gain matrix

GG	Gravitational acceleration, 32 ft/sec^2
GAMMA	Γ matrix
GNX,GNY,GNZ	G(1,1), G(4,2), and G(7.3) terms of the gain matrix
H	Measurement matrix
HT	Measurement matrix transposed
INVERT	Subroutine entry to invert a matrix
ITH	Equal to number of times through main processing loop, except on last time through when ITH = -1
IU	Seed for random number generator, NORMAL
NORMAL	Subroutine to generate normally distributed random variables to be used as noise
PE	Covariance of estimation error matrix
PH1	Estimated bank angle
PHC	Command bank angle
PHI	State transition matrix used for gain generation only
PHITRN	Transpose of PHI
PROD	Subroutine entry to perform the product of two matrices
PS1	Estimated heading angle with respect to the wind
PSN	Predicted new heading angle before acceleration estimate correction
PSD1	Estimated turning rate
Q	Array resulting from expected random forcing assumption
R	Range to target
RAN	Contains random numbers from NORMAL
SIG(1) - SIG(3)	Standard deviation for R, θ, and ϕ
SIG(4) - SIG(6)	Initial velocity estimates for prediction
SIGW	Standard deviations of random forcing in x,y, and z of radar frame
TB	Roll response time constant, τ_b
TRANS	Subroutine entry to perform the transpose of an array

VT1	Estimate of horizontal airspeed
VARR,VART,VARP	Variances of R, θ, and ϕ
WR	Estimated wind vector in the radar frame
X1,XD1,XDD1	Estimated state vector in radar frame (XDD1 not used in RADAR6)
X2,XD2	True state position and velocity in the radar frame
X1P,XD1P,XDD1P	Predicted state vector in radar frame
XDATA	Cartesian radar observation with noise
XIDENT	Identity matrix

D. COARSE GUIDANCE MAIN ROUTINE

ATITLE-ETITLE	Titles for plots
AZ	Array for storing average leg azimuths
AZI	Initial azimuth for first leg
BIS	Array for RADAR6 noise biases
DEG	Degree to radian conversion factor
DEL	Initial displacement vector for true aircraft position
DT	Set equal to DTCON
DTCON	Control update interval
DTRAD	Radar sampling interval
DTG	Distance from X1 to leg intersection
EEST	Estimated aircraft deviation from leg
ETRUE	True aircraft deviation from leg
FEET	Feet-to-nautical mile conversion factor
G	Gravitational acceleration constant
G1	HDE gain constant, G_1
G2	HDEDOT gain constant, G_2
H	Array of desired air headings
HEA	Estimated aircraft air heading
HTA	True aircraft air heading
HEG	Estimated aircraft ground heading

HTG	True aircraft ground heading
HDE	Heading angle error
HDEDOT	Heading angle error rate
HERMIN	Minimum error angle for which commands will be sent
I	Equal to the index on the current leg
IDTRAD	Integer relating DTCON and DTRAD
IEND	Equal to 0 except on last leg
ITH	Equal to the number of times through the main processing loop
IU	Seed for the random number generator
NLEG	Number of legs for the run
PHB	Bias angle; assumed zero
PHC	Command bank angle
PHD	Desired bank angle from controller
PLENGTH	Length of each leg
RAD	Radian to degree conversion factor
RMSEST	RMS value of EEST
RMSETR	RMS value of ETRUE
RANGE	Array containing average range to each leg
RANGEI	Range to start of first leg from the radar
SIG(1) - SIG(3)	Standard deviations for R, θ, and ϕ
SIG(4) - SIG(6)	Initial velocities for filter prediction
SIGW	Standard deviations for assumed random forcing; assumed zero for all runs in this study
T	Time into simulation run
TB	True aircraft roll response time constant
TAUH	Estimated aircraft roll response time constant; set equal to TB for all runs in this study
TINTRN	Time into a command turn
THETA	Array containing azimuths of each leg
THETW	Direction toward which true wind blows

THETWH	Direction toward which estimated wind blows
TLEG	Array containing desired time to cover each leg
TLEG1	Estimated time remaining on a leg
TSTOP	Time at which problem ends
TSTPTN	Time at which a zero bank command is sent to end a turn
TTOTTN	Time at which a turn is completely over
TG	Time to go to start a turn
VA	True airspeed
VAH	Estimated airspeed
VEG	Estimated ground speed
VTG	True ground speed
VW	True wind speed
VWH	Estimated wind speed
WR	Equal to WH; components of estimated wind
WT	Components of true wind
X1,XD1	Estimated aircraft position and velocity
X2,XD2	Desired track data; used for plotting only
X3,XD3	True aircraft position and velocity
XWP	Array containing x waypoint co-ordinates
XXA-XXE	Arrays for use in plotting
YYA-YYE	Arrays for use in plotting
YWP	Array containing y waypoint co-ordinates

Note: There are several variables with the suffix NM. These variables correspond to the variable without the suffix, except the NM indicates the value stored is in nautical miles.

APPENDIX B: PRECISION GUIDANCE INPUT DATA FORMAT DEFINITION

CARD	COLUMNS	FORMAT	VARIABLE	REMARKS
1	1-10	F10.3	WT(1)	True wind in target frame
	11-20		WT(2)	x,y, and z directions;WT(3)
	21-30		WT(3)	is usually zero
	31-40		WH(1)	Estimated wind in target
	41-50		WH(2)	frame;WH(3) is usually
	51-60		WH(3)	zero
2	1-10		BIS(1)	Bias on range, azimuth
	11-20		BIS(2)	and elevation noise
	21-30		BIS(3)	
	31-40		SIG(4)	Initial velocity estimates
	41-50		SIG(5)	for filter prediction
	51-60		SIG(6)	
3	1-10		SIG(1)	Standard deviations on
	11-20		SIG(2)	range, azimuth, and
	21-30		SIG(3)	elevation noise
4	1-10		SIGW(1)	Standard deviations of
	11-20		SIGW(2)	assumed random forcing in
	21-30		SIGW(3)	x,y,and z of radar frame
5	1-10		X3(1)	Initial position (x,y,z)
	11-20		X3(2)	of aircraft in target
	21-30		X3(3)	frame
7	1-10		PHB	Autopilot bias angle
	11-20		TB	Roll response time constant
	21-30		DT	Radar sampling interval
	31-40		DTCON	Control update interval
	41-50		PHI	Latitude of target
	51-60	F10.3	TWLD	Time to start wild points
	61-63	I3	MWLD	Number of wild points

CARD	COLUMNS	FORMAT	VARIABLE	REMARKS
8	1-10	F10.3	T3	Nominal dive duration
	11-20		THNM	Combine to give angle
	21-30		THDM	radar and target
	31-40		THD	Nominal dive angle
	41-50		ATT	True tang. accel. during dive
	51-60		ATH	Est. tang. accel. during dive
9	1-10		ANH	Max. aircraft normal accel.
	11-20		TUP	Aircraft pitch time constant
	21-30		HTOL	Angle and altitude tolerances
	31-40		ATOL	for dive pullout
	41-50	F10.3	S	Equal to 9999
	51-53	I3	IB1	Equal to zero for level flight
	54-56	I3	IACC	Equal to zero for no acceleration compensation
40-49	1-48	6A8	ATITLE-ZTITLE	Titles for plots. Two cards per plot for two lines of title per plot
50	1-10	F10.3	WBLX	Ballistic wind in x
	11-20		WBLY	Ballistic wind in y
	21-30		BFF	Ballistic form factor
	31-40		D	Bomb diameter, inches
	41-50	F10.3	W	Bomb weight, pounds
	51-53	I3	NTB	Number of bombing table to be used
51-...				Bombing table data as supplied with original program; this remains unchanged.

APPENDIX C: COARSE GUIDANCE INPUT DATA FORMAT DEFINITION

CARD	COLUMNS	FORMAT	VARIABLE	REMARKS
1	1	I1	NLEG	Number of legs
	2-10	F9.2	RANGEI	Range of first leg from radar
	11-20	F10.4	AZI	Azimuth of first leg from radar
	21-30		VA	True airspeed
	31-40		VAH	Est. airspeed
	41-50		VW	True wind speed
	51-60		VWH	Est. wind speed
	61-70		THETW	True direction of wind
	71-80		THETWH	Est. direction of wind
2	1-10	F10.3	PLENGTH(i)	Length of the ith leg and its associated azimuth
thru NLEG+1	11-20		THETA(i)	where i=1, NLEG
NLEG+2	1-10		HTG	True initial ground heading
	11-20		DELNM(1)	Displacement of true position from RANGEI, AZI in x and y
	21-30		DELNM(2)	
	31-40		DELNM(3)	DELNM(3) is altitude of aircraft in NM
NLEG+3 NLEG+4	1-48	6A8	ATITLE	Title for plot of true, estimated and desired paths

Note: All angles are in degrees from North, clockwise angles positive 0 - 360 degrees. All speeds in ft/sec. All lengths are in nautical miles on entry.

```
                         AN/TPQ-27 SIMULATION
              PRECISION GUIDANCE MODE WITH KALMAN FILTERING

INITIAL CONDITIONS :

   TRUE WIND AT TARGET =                  0.0          0.0          0.0
   ESTIMATED WIND AT TARGET =             0.0          0.0          0.0

   RADAR DATA
     NUMBER OF NOISY POINTS(MWLD) =       0
     START TIME OF WILD POINTS =          0.0
     MEASUREMENT SIGMAS(R(FT.),AZ(MRAD),EL(MRAD)) =   1.500000 01   1.000000-01   1.000000-01
     MEASUREMENT BIASES(             "            ) =   0.0           0.0           0.0
     INITIAL VELOCITY MEASUREMENT VALUES =             0.0           0.0           0.0
     RANDOM FORCING ASSUMPTION VALUES (SIGW) =         0.0           0.0           0.0
     RADAR SAMPLING INTERVAL =                         0.1250

   INITIAL POSITION OF A/C IN TARGET SYSTEM =    -3.000000 04   5.000000 03   1.000000 04
   INITIAL VELOCITY OF A/C IN TARGET SYSTEM =     5.000000 02   0.0           0.0


AIRCRAFT PARAMETERS :

    TB =  2.00000      PHB =  0.0


CONTROL PARAMETERS:

    DTCON =   0.12500
    G1 =  75.0000
    G2 =  75.0000


BALLISTIC TABLE PARAMETERS:

    BALLISTIC WIND VALUES (WBLX,WBLY) =   0.0       0.0
    PFF =    1.000000
    D =      4.000000
    W =     23.600000
    NTR =    144


DIVE BOMBING MODE PARAMETERS:

    THO  =     0.0        ATT  =    1.00
    ATH  =     1.00       ANH  =    1.00
    TUP  =     1.00       HTOL =    1.00
   ATOL  =     1.00       THNM =    5.00
   THDM  =     6.00       IB1  =    0
   IACC  =     0
```

H ARRAY

1.00000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	1.00000	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	1.00000	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

GAMMA ARRAY

0.00033	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.00781	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.12500	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.00033	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.00781	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.12500	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.00033	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.00781	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.12500	0.0	0.0	0.0	0.0	0.0	0.0

C ARRAY

0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

PP ARRAY

1000000.00000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	1000000.00000	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	1000000.00000	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	1000000.00000	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	1000000.00000	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	1000000.00000	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	1000000.00000	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	1000000.00000	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1000000.00000

PHI ARRAY

1.0000	0.1250	0.0078	0.0	0.0	0.0	0.0	0.0	0.0
0.0	1.0000	0.1250	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	1.0000	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	1.0000	0.1250	0.0078	0.0	0.0	0.0
0.0	0.0	0.0	0.0	1.0000	0.1250	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	1.0000	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	1.0000	0.1250	0.0078
0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0000	0.1250
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0000

R ARRAY

959.99368	71.15830	1.76608	0.0	0.0	0.0	0.0	0.0	0.0
71.15830	1181.25355	-18.23060	0.0	0.0	0.0	0.0	0.0	0.0
1.76608	-18.23060	967.02410	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

OUTPUT FORMAT

X1	Y1	Z1	XD1	YD1	ZD1	XDD1	YDD1
X2	Y2	Z2	XD2	YD2	ZD2	TF	RA
X3	Y3	Z3	XD3	YD3	ZD3	XGC	YGC
X5	Y5	Z5	XD5	YD5	ZD5	AA	HAT
X6	Y6	Z6	XD6	YD6	ZD6	YNP	
PHD1	PHD2	PHD	HDE	HDEDOT	TG	VEH	THN
PH	PHI	PHC	PS	PSI			
XE	DXE	CTV		G(1,1)	G(4,2)	G(7,3)	T
-0.29982D 05	0.30950D 06	0.76753D 04	0.0	0.0	0.0	0.0	0.0
-0.30000D 05	0.30950D 06	0.77129D 04	0.50000D 03	0.0	0.0	0.0	0.0
-0.30000D 05	0.50000D 04	0.10000D 05	0.50000D 03	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.30155D-79	-0.14737D-57
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.10000D 04	0.85914D-80	0.0
0.0	0.0	0.0	0.90000D 02	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.99904D 00	0.99882D 00	0.99903D 00	0.0
-0.29512D 05	0.30951D 06	0.76900D 04	0.46070D 03	0.45715D 02	-0.51560D 02	-0.10322D 03	0.56847D 02
-0.29500D 05	0.30950D 06	0.77129D 04	0.50000D 03	0.17437D-12	-0.25364D-14	0.0	0.0
-0.29500D 05	0.50000D 04	0.10000D 05	0.50000D 03	0.17439D-12	0.0	0.0	0.0
0.29512D 05	-0.50146D 04	-0.99774D 04	0.46070D 03	0.46460D 02	-0.50890D 02	0.84241D 02	0.99774D 04
0.79505D 04	0.28860D 05	-0.99774D 04	-0.71054D-14	0.46303D 03	-0.50890D 02	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.10000D 04	0.46303D 03	0.0
0.0	0.0	0.0	0.90000D 02	0.84333D 02	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.64524D 00	0.64194D 00	0.64514D 00	0.10000D 01
-0.29020D 05	0.30948D 06	0.77194D 04	0.47374D 03	-0.24265D 02	-0.79172D 01	-0.25066D 02	-0.18723D 02
-0.29000D 05	0.30950D 06	0.77129D 04	0.50000D 03	0.17437D-12	-0.25364D-14	0.0	0.0
-0.29000D 05	0.50000D 04	0.10000D 05	0.50000D 03	0.17439D-12	0.0	0.0	0.0
0.29020D 05	-0.49845D 04	-0.10006D 05	0.47374D 03	-0.24147D 02	-0.82693D 01	0.92918D 02	0.10006D 05
0.35007D 04	0.29237D 05	-0.10006D 05	0.0	0.47435D 03	-0.82693D 01	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.10000D 04	0.47435D 03	0.0
0.0	0.0	0.0	0.90000D 02	0.92932D 02	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.42141D 00	0.42124D 00	0.42144D 00	0.20000D 01
-0.28502D 05	0.30950D 06	0.77156D 04	0.50268D 03	0.54558D 01	-0.15093D 02	-0.10303D 01	0.36094D 01
-0.28500D 05	0.30950D 06	0.77129D 04	0.50000D 03	0.17437D-12	-0.25364D-14	0.0	0.0
-0.28500D 05	0.50000D 04	0.10000D 05	0.50000D 03	0.17439D-12	0.0	0.0	0.0
0.28502D 05	-0.50002D 04	-0.10003D 05	0.50268D 03	0.56747D 01	-0.15012D 02	0.39353D 02	0.10003D 05
0.53216D 04	0.28444D 05	-0.10003D 05	0.0	0.50271D 03	-0.15012D 02	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.10000D 04	0.50271D 03	0.0
0.0	0.0	0.0	0.90000D 02	0.89378D 02	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.30792D 00	0.30788D 00	0.30796D 00	0.30000D 01
-0.27999D 05	0.30949D 06	0.77197D 04	0.50360D 03	-0.40607D 00	-0.11908D 02	-0.45855D-01	-0.44679D-01
-0.28000D 05	0.30950D 06	0.77129D 04	0.50000D 03	0.17437D-12	-0.25364D-14	0.0	0.0
-0.28000D 05	0.50000D 04	0.10000D 05	0.50000D 03	0.17439D-12	0.0	0.0	0.0
0.27999D 05	-0.49966D 04	-0.10007D 05	0.50360D 03	-0.23284D 00	-0.11912D 02	0.90026D 02	0.10007D 05
0.49837D 04	0.28001D 05	-0.10007D 05	0.0	0.50360D 03	-0.11912D 02	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.10000D 04	0.50360D 03	0.0
0.0	0.0	0.0	0.90000D 02	0.90046D 02	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.24213D 00	0.24211D 00	0.24217D 00	0.40000D 01
-0.27506D 05	0.30951D 06	0.77122D 04	0.49375D 03	0.98639D 01	-0.14778D 02	-0.34462D 01	0.30328D 01
-0.27500D 05	0.30950D 06	0.77129D 04	0.50000D 03	0.17437D-12	-0.25364D-14	0.0	0.0
-0.27500D 05	0.50000D 04	0.10000D 05	0.50000D 03	0.17439D-12	0.0	0.0	0.0
0.27506D 05	-0.50119D 04	-0.99995D 04	0.49375D 03	0.10078D 02	-0.14633D 02	0.88831D 02	0.99995D 04
0.55721D 04	0.27398D 05	-0.99995D 04	0.0	0.49385D 03	-0.14633D 02	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.10000D 04	0.49385D 03	0.0
0.0	0.0	0.0	0.90000D 02	0.88856D 02	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.19941D 00	0.19938D 00	0.19945D 00	0.50000D 01
-0.27009D 05	0.30950D 06	0.77191D 04	0.49373D 03	0.36041D 01	-0.84222D 01	-0.25186D 01	0.63886D 00
-0.27000D 05	0.30950D 06	0.77129D 04	0.50000D 03	0.17437D-12	-0.25364D-14	0.26956D 02	0.10853D 05
-0.27000D 05	0.50000D 04	0.10000D 05	0.50000D 03	0.17439D-12	0.0	0.14844D 00	0.10873D 05
-0.27009D 05	0.50072D 04	0.10006D 05	0.49373D 03	0.37262D 01	-0.83689D 01	0.89568D 02	0.10006D 05
0.87962D 01	-0.72443D 01	-0.63331D 01	0.62714D 01	-0.37262D 01	0.83689D 01	0.0	0.0
0.82013D 03	0.0	0.82013D 03	0.10935D 02	0.0	0.32603D 02	0.49374D 03	0.0
0.0	0.0	0.29883D 02	0.90000D 02	0.89582D 02	0.0	0.0	0.0
0.52108D 04	0.0	-0.93661D 02	0.0	0.16945D 00	0.16943D 00	0.16949D 00	0.60000D 01

OUTPUT FORMAT

X1 X2 X3 X5 X6 PHD1 PH XE	Y1 Y2 Y3 Y5 Y6 PHD2 PHI DXE	Z1 Z2 Z3 Z5 Z6 PHD PHC CTV	XD1 XD2 XD3 XD5 XD6 HDF PS	YD1 YD2 YD3 YD5 YD6 HDEDOT PS1 G(1,1)	ZD1 ZD2 ZD3 ZD5 ZD6 TG G(4,2)	XDD1 TF XGC AA YNP VEH G(7,3)	YDD1 RA YGC HAT THN T
-0.23011D 05	0.30917D 06	0.77107D 04	0.49375D 03	-0.95321D 02	-0.71336D 00	0.29717D 00	0.13734D 00
-0.23021D 05	0.30916D 06	0.77177D 04	0.49059D 03	-0.96528D 02	0.14040D 01	0.26955D 02	0.11025D 05
-0.23021D 05	0.46682D 04	0.10000D 05	0.49059D 03	-0.96539D 02	0.0	-0.37226D 01	0.11044D 05
-0.23011D 05	0.46719D 04	0.99931D 04	0.49376D 03	-0.95301D 02	-0.20996D 01	0.10092D 03	0.99931D 04
-0.10250D 02	-0.36845D 01	0.69031D 01	-0.31632D 01	-0.12376D 01	0.20996D 01	0.0	0.0
0.42083D 02	-0.64065D 02	-0.21982D 02	0.56110D 00	-0.85420D 00	0.24729D 02	0.50287D 03	0.0
0.13599D 02	0.13599D 02	-0.22031D 02	0.10113D 03	0.10093D 03	0.0	0.0	0.0
0.22995D 03	-0.35589D 03	-0.49246D 01	0.0	0.76950D-01	0.76[illegible]20D-01	0.76995D-01	0.14000D 02
-0.22519D 05	0.30907D 06	0.77129D 04	0.49302D 03	-0.10015D 03	-0.65565D-01	0.25445D 00	-0.17706D-01
-0.22531D 05	0.30907D 06	0.77191D 04	0.48988D 03	-0.10006D 03	0.14554D 01	0.26957D 02	0.11030D 05
-0.22531D 05	0.45692D 04	0.10000D 05	0.48983D 03	-0.10007D 03	0.0	-0.29105D 01	0.11049D 05
-0.22519D 05	0.45685D 04	0.99938D 04	0.49302D 03	-0.10014D 03	-0.15222D 01	0.10148D 03	0.99938D 04
-0.12651D 02	0.69047D 00	0.62077D 01	-0.31339D 01	0.69201D-01	0.15222D 01	0.0	0.0
-0.26803D 00	-0.30583D 02	-0.30851D 02	-0.35737D-02	-0.40777D 00	0.23711D 02	0.50308D 03	0.0
0.10878D 01	0.10878D 01	-0.30000D 02	0.10155D 03	0.10148D 03	0.0	0.0	0.0
-0.14332D 01	-0.16393D 03	0.31379D-01	0.0	0.72042D-01	0.72011D-01	0.72085D-01	0.15000D 02
-0.22038D 05	0.30897D 06	0.77153D 04	0.49031D 03	-0.99669D 02	0.49847D 00	-0.42301D-01	0.57397D-01
-0.22042D 05	0.30897D 06	0.77206D 04	0.48983D 03	-0.10031D 03	0.14591D 01	0.26953D 02	0.10976D 05
-0.22042D 05	0.44690D 04	0.10000D 05	0.48963D 03	-0.10032D 03	0.0	-0.39130D 01	0.10995D 05
-0.22038D 05	0.44714D 04	0.99947D 04	0.49031D 03	-0.99665D 02	-0.95115D 00	0.10149D 03	0.99947D 04
-0.37729D 01	-0.24288D 01	0.52975D 01	-0.47399D 00	-0.65564D 00	0.95115D 00	0.0	0.0
-0.79422D 00	0.34415D 01	0.26472D 01	-0.10590D-01	0.45896D-01	0.22968D 02	0.50033D 03	0.0
-0.82275D 00	-0.82275D 00	0.26953D 01	0.10157D 03	0.10149D 03	0.0	0.0	0.0
-0.41561D 01	0.18147D 02	0.92474D-01	0.0	0.67723D-01	0.67691D-01	0.67765D-01	0.16000D 02
-0.21542D 05	0.30887D 06	0.77117D 04	0.49154D 03	-0.99652D 02	-0.35893D 00	0.84361D-01	0.67743D-01
-0.21552D 05	0.30887D 06	0.77221D 04	0.48980D 03	-0.10047D 03	0.14613D 01	0.26948D 02	0.10997D 05
-0.21552D 05	0.43686D 04	0.10000D 05	0.48980D 03	-0.10048D 03	0.0	-0.38958D 01	0.11016D 05
-0.21542D 05	0.43724D 04	0.99897D 04	0.49154D 03	-0.99636D 02	-0.18082D 01	0.10146D 03	0.99897D 04
-0.99362D 01	-0.37587D 01	0.10265D 02	-0.17427D 01	-0.84166D 00	0.18032D 01	0.0	0.0
0.18796D 01	0.53082D 01	0.71873D 01	0.25061D-01	0.70776D-01	0.21862D 02	0.50154D 03	0.0
0.82057D-01	0.82057D-01	0.71484D 01	0.10159D 03	0.10146D 03	0.0	0.0	0.0
0.96144D 01	0.27005D 02	-0.21937D 00	0.0	0.63890D-01	0.63856D-01	0.63934D-01	0.17000D 02
-0.21053D 05	0.30877D 06	0.77150D 04	0.49114D 03	-0.99851D 02	0.39059D 00	0.32157D-01	0.15002D-01
-0.21062D 05	0.30876D 06	0.77235D 04	0.48987D 03	-0.10014D 03	0.14566D 01	0.26950D 02	0.10991D 05
-0.21062D 05	0.42682D 04	0.10000D 05	0.48987D 03	-0.10016D 03	0.0	-0.39116D 01	0.11010D 05
-0.21053D 05	0.42700D 04	0.99915D 04	0.49114D 03	-0.99846D 02	-0.10617D 01	0.10149D 03	0.99915D 04
-0.93055D 01	-0.17518D 01	0.85174D 01	-0.12712D 01	-0.30881D 00	0.10617D 01	0.0	0.0
-0.11621D 01	-0.93954D 01	-0.10558D 02	-0.15494D-01	-0.12527D 00	0.20893D 02	0.50118D 03	0.0
0.67533D-01	0.67533D-01	-0.10547D 02	0.10156D 03	0.10149D 03	0.0	0.0	0.0
-0.58090D 01	-0.46966D 02	0.13553D 00	0.0	0.60473D-01	0.60439D-01	0.60513D-01	0.18000D 02
-0.20568D 05	0.30867D 06	0.77229D 04	0.49002D 03	-0.99318D 02	0.14329D 01	-0.80905D-01	-0.27461D-02
-0.20572D 05	0.30867D 06	0.77250D 04	0.49002D 03	-0.99412D 02	0.14460D 01	0.26957D 02	0.10971D 05
-0.20572D 05	0.41685D 04	0.10000D 05	0.49002D 03	-0.99423D 02	0.0	-0.39046D 01	0.10990D 05
-0.20569D 05	0.41693D 04	0.99979D 04	0.49002D 03	-0.99336D 02	0.48818D 00	0.10146D 03	0.99979D 04
-0.43736D 01	-0.83104D 00	0.20781D 01	-0.60758D-02	-0.87399D-01	-0.48818D 00	0.0	0.0
0.77830D 00	0.30388D 02	0.31166D 02	0.10377D-01	0.40517D 00	0.19992D 02	0.49999D 03	0.0
-0.22920D 01	-0.22920D 01	0.29883D 02	0.10147D 03	0.10146D 03	0.0	0.0	0.0
0.38009D 01	0.14874D 03	-0.90557D-01	0.0	0.57399D-01	0.57365D-01	0.57440D-01	0.19000D 02
-0.20075D 05	0.30857D 06	0.77231D 04	0.49040D 03	-0.99560D 02	0.17343D 01	-0.43384D-01	-0.19218D-01
-0.20082D 05	0.30957D 06	0.77264D 04	0.49001D 03	-0.99450D 02	0.14465D 01	0.26956D 02	0.10978D 05
-0.20082D 05	0.40691D 04	0.10000D 05	0.49001D 03	-0.99461D 02	0.0	-0.35101D 01	0.10998D 05
-0.20075D 05	0.40690D 04	0.99967D 04	0.49040D 03	-0.99574D 02	0.28614D 00	0.10143D 03	0.99967D 04
-0.68446D 01	0.97033D-01	0.32873D 01	-0.39272D 00	0.11369D 00	-0.28614D 00	0.0	0.0
-0.65837D 00	-0.10607D 02	-0.11266D 02	-0.87783D-02	-0.14143D 00	0.18956D 02	0.50041D 03	0.0
0.10551D 00	0.10551D 00	-0.11250D 02	0.10147D 03	0.10148D 03	0.0	0.0	0.0
-0.51383D 01	-0.50640D 02	0.76668D-01	0.0	0.54623D-01	0.54589D-01	0.54664D-01	0.20000D 02

OUTPUT FORMAT

X1 X2 X3 X5 X6 PHD1 PH XE	Y1 Y2 Y3 Y5 Y6 PHD2 PHI DXE	Z1 Z2 Z3 Z5 Z6 PHD PHC CTV	XD1 XD2 XD3 XD5 XD6 HOE PS	YD1 YD2 YD3 YD5 YD6 HOEDC? PS1 G(1,1)	ZD1 ZD2 ZD3 ZD5 ZD6 TG G(4,2)	XDD1 TF XGC AA YNP VEH G(7,3)	YDD1 RA YGC HAT THN T
-0.19589D 05	0.30847D 06	0.77264D 04	0.48955D 03	-0.99846D 02	0.21554D 01	-0.10496D 00	-0.27335D-01
-0.19592D 05	0.30847D 06	0.77279D 04	0.48997D 03	-0.99621D 02	0.14490D 01	0.26957D 02	0.10964D 05
-0.19592D 05	0.39696D 04	0.10000D 05	0.48997D 03	-0.99631D 02	0.0	-0.39286D 01	0.10983D 05
-0.19589D 05	0.39689D 04	0.99986D 04	0.48955D 03	-0.99867D 02	0.70301D 00	0.10153D 03	0.99986D 04
-0.31835D 01	0.69352D 00	0.14392D 01	0.42457D 00	0.23583D 00	-0.70301D 00	0.0	0.0
-0.48776D 01	-0.36189D 02	-0.41066D 02	-0.65035D-01	-0.48251D 00	0.18020D 02	0.49963D 03	0.0
-0.15960D 00	-0.15960D 00	-0.30000D 02	0.10149D 03	0.10153D 03	0.0	0.0	0.0
-0.22686D 02	-0.16828D 03	0.56712D 00	0.0	0.52106D-01	0.52072D-01	0.52145D-01	0.21000D 02
-0.19094D 05	0.30837D 06	0.77176D 04	0.49042D 03	-0.99070D 02	0.36763D 00	-0.39233D-01	-0.53149D-02
-0.19102D 05	0.30837D 06	0.77293D 04	0.49008D 03	-0.99108D 02	0.14416D 01	0.26943D 02	0.10972D 05
-0.19102D 05	0.38704D 04	0.10000D 05	0.49008D 03	-0.99118D 02	0.0	-0.38857D 01	0.10992D 05
-0.19094D 05	0.38712D 04	0.99883D 04	0.49042D 03	-0.99065D 02	-0.10733D 01	0.10142D 03	0.99883D 04
-0.74823D 01	-0.81379D 00	0.11709D 02	-0.33956D 00	-0.53384D-01	0.10733D 01	0.0	0.0
0.38972D 01	-0.24445D 01	0.14526D 01	0.51962D-01	-0.32594D-01	0.16972D 02	0.50032D 03	0.0
0.20203D 01	0.20203D 01	0.14063D 01	0.10143D 03	0.10142D 03	0.0	0.0	0.0
0.17669D 02	-0.11583D 02	-0.45375D 00	0.0	0.49808D-01	0.49773D-01	0.49849D-01	0.22000D 02
-0.18608D 05	0.30827D 06	0.77333D 04	0.48958D 03	-0.99629D 02	0.29839D 01	-0.84841D-01	0.10817D-01
-0.18612D 05	0.30827D 06	0.77308D 04	0.48992D 03	-0.99895D 02	0.14530D 01	0.26963D 02	0.10966D 05
-0.18612D 05	0.37707D 04	0.10000D 05	0.48992D 03	-0.99905D 02	0.0	-0.39230D 01	0.10985D 05
-0.18608D 05	0.37728D 04	0.10003D 05	0.48958D 03	-0.99662D 02	0.15346D 01	0.10151D 03	0.10003D 05
-0.40084D 01	-0.21026D 01	-0.25812D 01	0.33908D 00	-0.24329D 00	-0.15346D 01	0.0	0.0
-0.24750D 01	0.80756D 00	-0.16674D 01	-0.33000D-01	0.10768D-01	0.16015D 02	0.49962D 03	0.0
-0.98708D 00	-0.98708D 00	-0.16406D 01	0.10153D 03	0.10150D 03	0.0	0.0	0.0
-0.10935D 02	0.38636D 01	0.28776D 00	0.0	0.47708D-01	0.47673D-01	0.47747D-01	0.23000D 02
-0.18112D 05	0.30817D 06	0.77399D 04	0.49055D 03	-0.99592D 02	0.37824D 01	-0.11940D-01	0.18652D-01
-0.18122D 05	0.30817D 06	0.77322D 04	0.48990D 03	-0.99982D 02	0.14543D 01	0.26972D 02	0.10987D 05
-0.18122D 05	0.36709D 04	0.10000D 05	0.48990D 03	-0.99993D 02	0.0	-0.39181D 01	0.11006D 05
-0.18112D 05	0.36739D 04	0.10008D 05	0.49055D 03	-0.99637D 02	0.23335D 01	0.10148D 03	0.10008D 05
-0.10319D 02	-0.29398D 01	-0.77872D 01	-0.64869D 00	-0.35604D 00	-0.23335D 01	0.0	0.0
-0.19540D 00	-0.17711D 02	-0.17906D 02	-0.26053D-02	-0.23615D 00	0.14932D 02	0.50056D 03	0.0
0.10868D 01	0.10668D 01	-0.17920D 02	0.10154D 03	0.10148D 03	0.0	0.0	0.0
-0.84033D 00	-0.76404D 02	0.22761D-01	0.0	0.45776D-01	0.45741D-01	0.45815D-01	0.24000D 02
-0.17618D 05	0.30807D 06	0.77431D 04	0.49103D 03	-0.99536D 02	0.39035D 01	0.20149D-01	0.18676D-01
-0.17632D 05	0.30807D 06	0.77337D 04	0.48991D 03	-0.99944D 02	0.14537D 01	0.26976D 02	0.10997D 05
-0.17632D 05	0.35709D 04	0.10000D 05	0.48991D 03	-0.99954D 02	0.0	-0.39135D 01	0.11016D 05
-0.17618D 05	0.35742D 04	0.10010D 05	0.49103D 03	-0.99582D 02	0.24554D 01	0.10146D 03	0.10010D 05
-0.13996D 02	-0.33488D 01	-0.95141D 01	-0.11199D 01	-0.37228D 00	-0.24554D 01	0.0	0.0
0.12176D 01	0.12843D 02	0.14061D 02	0.16235D-01	0.17124D 00	0.13893D 02	0.50102D 03	0.0
-0.23330D 00	-0.23330D 00	0.14063D 02	0.10153D 03	0.10146D 03	0.0	0.0	0.0
0.50937D 01	0.53774D 02	-0.14196D 00	0.0	0.43995D-01	0.43960D-01	0.44033D-01	0.25000D 02
-0.17123D 05	0.30797D 06	0.77439D 04	0.49169D 03	-0.99434D 02	0.36470D 01	0.63460D-01	0.37146D-01
-0.17142D 05	0.30797D 06	0.77351D 04	0.48987D 03	-0.10015D 03	0.14567D 01	0.26976D 02	0.11009D 05
-0.17142D 05	0.34709D 04	0.10000D 05	0.48987D 03	-0.10016D 03	0.0	-0.39042D 01	0.11028D 05
-0.17123D 05	0.34765D 04	0.10009D 05	0.49169D 03	-0.99477D 02	0.22004D 01	0.10144D 03	0.10009D 05
-0.19624D 02	-0.56046D 01	-0.88289D 01	-0.18220D 01	-0.68058D 00	-0.22004D 01	0.0	0.0
0.39287D 01	0.65998D 00	0.45887D 01	0.52383D-01	0.87999D-02	0.12845D 02	0.50165D 03	0.0
0.21751D 01	0.21751D 01	0.45703D 01	0.10156D 03	0.10143D 03	0.0	0.0	0.0
0.15974D 02	0.22404D 01	-0.45863D 00	0.0	0.42349D-01	0.42314D-01	0.42386D-01	0.26000D 02
-0.16634D 05	0.30787D 06	0.77448D 04	0.49130D 03	-0.99702D 02	0.34407D 01	0.36572D-01	0.20571D-01
-0.16652D 05	0.30787D 06	0.77366D 04	0.48986D 03	-0.10018D 03	0.14571D 01	0.26975D 02	0.11002D 05
-0.16652D 05	0.33706D 04	0.10000D 05	0.48986D 03	-0.10019D 03	0.0	-0.39168D 01	0.11021D 05
-0.16634D 05	0.33750D 04	0.10008D 05	0.49130D 03	-0.99742D 02	0.19902D 01	0.10148D 03	0.10008D 05
-0.18369D 02	-0.44059D 01	-0.83356D 01	-0.14415D 01	-0.44596D 00	-0.19902D 01	0.0	0.0
0.50223D 00	0.68027D 01	0.73049D 01	0.66965D-02	0.90702D-01	0.11871D 02	0.50132D 03	0.0
-0.18156D 00	-0.18156D 00	0.72656D 01	0.10156D 03	0.10147D 03	0.0	0.0	0.0
0.19837D 01	0.26910D 02	-0.58592D-01	0.0	0.40821D-01	0.40786D-01	0.40858D-01	0.27000D 02

OUTPUT FORMAT

```
        X1            Y1            Z1            XD1           YD1           ZD1           XDD1          YDD1
        X2            Y2            Z2            XD2           YD2           ZD2             TF            RA
        X3            Y3            Z3            XD3           YD3           ZD3            XGC           YGC
        X5            Y5            Z5            XD5           YD5           ZD5             AA           HAT
        X6            Y6            Z6            XD6           YD6           ZD6            YNP
      PHD1          PHD2           PHD            HDE        HDEDCT            TG            VEH           THN
        PH           PH1           PHC             PS           PSI
        XE           DXE           CTV                       G(1,1)        G(4,2)         G(7,3)             T

-0.16149D 05  0.30777D 06  0.77465D 04  0.49042D 03 -0.99618D 02  0.33505D 01 -0.19466D-01  0.15658D-01
-0.16162D 05  0.30777D 06  0.77380D 04  0.48989D 03 -0.10003D 03  0.14550D 01  0.26973D 02  0.10985D 05
-0.16162D 05  0.32704D 04  0.10000D 05  0.48989D 03 -0.10004D 03  0.0         -0.39202D 01  0.11004D 05
-0.16149D 05  0.32747D 04  0.10009D 05  0.49042D 03 -0.99656D 02  0.19014D 01  0.10149D 03  0.10009D 05
-0.13093D 02 -0.43078D 01 -0.85609D 01 -0.52889D 00 -0.38533D 00 -0.19014D 01  0.0          0.0
-0.76109D 00  0.13911D 02  0.13149D 02 -0.10148D-01  0.18547D 00  0.10938D 02  0.50044D 03  0.0
-0.14407D 01 -0.14407D 01  0.13125D 02  0.10154D 03  0.10148D 03  0.0          0.0          0.0
-0.29184D 01  0.53631D 02  0.88634D-01  0.0          0.39400D-01  0.39365D-01  0.39436D-01  0.28000D 02

-0.15659D 05  0.30767D 06  0.77457D 04  0.49041D 03 -0.99521D 02  0.29450D 01 -0.19295D-01  0.18552D-01
-0.15673D 05  0.30767D 06  0.77395D 04  0.48990D 03 -0.10000D 03  0.14545D 01  0.26970D 02  0.10983D 05
-0.15673D 05  0.31704D 04  0.10000D 05  0.48990D 03 -0.10001D 03  0.0         -0.39150D 01  0.11002D 05
-0.15659D 05  0.31756D 04  0.10006D 05  0.49041D 03 -0.99553D 02  0.14973D 01  0.10148D 03  0.10006D 05
-0.13668D 02 -0.51279D 01 -0.62982D 01 -0.51398D 00 -0.45733D 00 -0.14973D 01  0.0          0.0
 0.21053D 00  0.48408D 01  0.50513D 01  0.28070D-02  0.64544D-01  0.99422D 01  0.50041D 03  0.0
-0.24036D 00 -0.24036D 00  0.50391D 01  0.10154D 03  0.10147D 03  0.0          0.0          0.0
 0.78277D 00  0.18045D 02 -0.24516D-01  0.0          0.38075D-01  0.38039D-01  0.38110D-01  0.29000D 02

-0.15177D 05  0.30757D 06  0.77430D 04  0.48942D 03 -0.99332D 02  0.29878D 01 -0.78729D-01 -0.14148D-02
-0.15183D 05  0.30757D 06  0.77409D 04  0.49000D 03 -0.99468D 02  0.14468D 01  0.26969D 02  0.10964D 05
-0.15183D 05  0.30706D 04  0.10000D 05  0.49000D 03 -0.99478D 02  0.0         -0.39158D 01  0.10983D 05
-0.15177D 05  0.30734D 04  0.10007D 05  0.48942D 03 -0.99365D 02  0.15428D 01  0.10148D 03  0.10007D 05
-0.58748D 01 -0.28379D 01 -0.71356D 01  0.58810D 00 -0.11271D 00 -0.15428D 01  0.0          0.0
-0.10579D 01  0.78867D 01  0.68287D 01 -0.14106D-01  0.10516D 00  0.90146D 01  0.49940D 03  0.0
-0.14119D 01 -0.14119D 01  0.67969D 01  0.10148D 03  0.10147D 03  0.0          0.0          0.0
-0.38123D 01  0.28650D 02  0.12295D 00  0.0          0.36837D-01  0.36802D-01  0.36871D-01  0.30000D 02

-0.14689D 05  0.30747D 06  0.77444D 04  0.48922D 03 -0.99237D 02  0.22588D 01 -0.86657D-01 -0.63324D-02
-0.14693D 05  0.30747D 06  0.77424D 04  0.49004D 03 -0.99279D 02  0.14441D 01  0.26961D 02  0.10957D 05
-0.14693D 05  0.29713D 04  0.10000D 05  0.49004D 03 -0.99290D 02  0.0         -0.39161D 01  0.10976D 05
-0.14689D 05  0.29736D 04  0.10002D 05  0.48922D 03 -0.99259D 02  0.81524D 00  0.10147D 03  0.10002D 05
-0.40230D 01 -0.23519D 01 -0.20647D 01  0.81786D 00 -0.31015D-01 -0.81524D 00  0.0          0.0
-0.71844D 00 -0.58500D 01 -0.65685D 01 -0.95791D-02 -0.78000D-01  0.80336D 01  0.49919D 03  0.0
-0.18625D 00 -0.18625D 00 -0.65625D 01  0.10145D 03  0.10147D 03  0.0          0.0          0.0
-0.25056D 01 -0.20404D 02  0.33459D-01  0.0          0.35675D-01  0.35641D-01  0.35710D-01  0.31000D 02

-0.14201D 05  0.30737D 06  0.77461D 04  0.48902D 03 -0.98978D 02  0.22570D 01 -0.94656D-01 -0.18487D-02
-0.14203D 05  0.30737D 06  0.77438D 04  0.49008D 03 -0.99101D 02  0.14415D 01  0.26961D 02  0.10952D 05
-0.14203D 05  0.28721D 04  0.10000D 05  0.49008D 03 -0.99112D 02  0.0         -0.39019D 01  0.10971D 05
-0.14201D 05  0.28752D 04  0.10002D 05  0.48902D 03 -0.99000D 02  0.81723D 00  0.10144D 03  0.10002D 05
-0.18151D 01 -0.30959D 01 -0.22845D 01  0.10630D 01 -0.11162D 00 -0.81723D 00  0.0          0.0
 0.12391D 01  0.36927D 01  0.49318D 01  0.16522D-01  0.49236D-01  0.70501D 01  0.49894D 03  0.0
 0.25394D 00  0.25394D 00  0.49219D 01  0.10143D 03  0.10144D 03  0.0          0.0          0.0
 0.41780D 01  0.12358D 02 -0.14387D 00  0.0          0.34586D-01  0.34551D-01  0.34620D-01  0.32000D 02

-0.13717D 05  0.30727D 06  0.77471D 04  0.48826D 03 -0.98926D 02  0.21784D 01 -0.12696D 00  0.37521D-02
-0.13712D 05  0.30727D 06  0.77453D 04  0.49007D 03 -0.99161D 02  0.14423D 01  0.26959D 02  0.10937D 05
-0.13712D 05  0.27729D 04  0.10000D 05  0.49007D 03 -0.99171D 02  0.0         -0.39053D 01  0.10956D 05
-0.13717D 05  0.27771D 04  0.10002D 05  0.48826D 03 -0.98947D 02  0.73939D 00  0.10145D 03  0.10002D 05
 0.47513D 01 -0.41843D 01 -0.19221D 01  0.18042D 01 -0.22385D 00 -0.73939D 00  0.0          0.0
 0.37850D 00  0.56547D 01  0.60332D 01  0.50467D-02  0.75396D-01  0.61004D 01  0.49819D 03  0.0
-0.22486D 00 -0.22486D 00  0.59766D 01  0.10144D 03  0.10145D 03  0.0          0.0          0.0
 0.12328D 01  0.18454D 02 -0.43881D-01  0.0          0.33562D-01  0.33527D-01  0.33594D-01  0.33000D 02

-0.13227D 05  0.30717D 06  0.77495D 04  0.48837D 03 -0.98997D 02  0.22736D 01 -0.11633D 00 -0.31420D-02
-0.13222D 05  0.30717D 06  0.77467D 04  0.49008D 03 -0.99093D 02  0.14414D 01  0.26961D 02  0.10940D 05
-0.13222D 05  0.26737D 04  0.10000D 05  0.49008D 03 -0.99104D 02  0.0         -0.39078D 01  0.10959D 05
-0.13227D 05  0.26769D 04  0.10003D 05  0.48837D 03 -0.99020D 02  0.83351D 00  0.10146D 03  0.10003D 05
 0.47238D 01 -0.31712D 01 -0.28730D 01  0.17128D 01 -0.83868D-01 -0.83351D 00  0.0          0.0
-0.30920D 00 -0.72706D 00 -0.10363D 01 -0.41227D-02 -0.96942D-02  0.50896D 01  0.49830D 03  0.0
-0.18564D 00 -0.18564D 00 -0.10547D 01  0.10143D 03  0.10146D 03  0.0          0.0          0.0
-0.97105D 00 -0.22580D 01  0.35855D-01  0.0          0.32596D-01  0.32562D-01  0.32628D-01  0.34000D 02
```

OUTPUT FORMAT

X1	Y1	Z1	XD1	YD1	ZD1	XDD1	YDD1
X2	Y2	Z2	XD2	YD2	ZD2	TF	RA
X3	Y3	Z3	XD3	YD3	ZD3	XGC	YGC
X5	Y5	Z5	XD5	YD5	ZD5	AA	HAT
X6	Y6	Z6	XD6	YD6	ZD6	YNP	
PHD1	PHD2	PHD	HDE	HDEDOT	TS	VEH	THN
PH	PH1	PHC	PS	PS1			
XE	DXE	CTV		G(1,1)	G(4,2)	G(7,3)	T
-0.12740D 05	0.30707D 06	0.77503D 04	0.48818D 03	-0.98821D 02	0.21658D 01	-0.12333D 00	-0.14183D-01
-0.12732D 05	0.30707D 06	0.77481D 04	0.49016D 03	-0.98678D 02	0.14353D 01	0.26959D 02	0.10935D 05
-0.12732D 05	0.25749D 04	0.10000D 05	0.49016D 03	-0.98689D 02	0.0	-0.39020D 01	0.10955D 05
-0.12740D 05	0.25760D 04	0.10002D 05	0.48818D 03	-0.98842D 02	0.72838D 00	0.10145D 03	0.10002D 05
0.78672D 01	-0.11625D 01	-0.22162D 01	0.19790D 01	0.15296D 00	-0.72838D 00	0.0	0.0
0.17357D 00	0.19434D 01	0.21170D 01	0.23142D-02	0.25912D-01	0.41025D 01	0.49809D 03	0.0
-0.26258D 00	-0.26258D 00	0.21094D 01	0.10138D 03	0.10144D 03	0.0	0.0	0.0
0.52499D 00	0.58865D 01	-0.20118D-01	0.0	0.31685D-01	0.31650D-01	0.31716D-01	0.35000D 02
-0.12248D 05	0.30697D 06	0.77542D 04	0.48856D 03	-0.98866D 02	0.24057D 01	-0.10142D 00	-0.20932D-01
-0.12242D 05	0.30697D 06	0.77496D 04	0.49019D 03	-0.98562D 02	0.14336D 01	0.26964D 02	0.10944D 05
-0.12242D 05	0.24762D 04	0.10000D 05	0.49019D 03	-0.98573D 02	0.0	-0.39026D 01	0.10963D 05
-0.12248D 05	0.24758D 04	0.10005D 05	0.48856D 03	-0.98691D 02	0.96758D 00	0.10144D 03	0.10005D 05
0.56016D 01	0.38472D 00	-0.46421D 01	0.16270D 01	0.31849D 00	-0.96758D 00	0.0	0.0
0.23939D 00	0.72512D 00	0.96451D 00	0.31918D-02	0.96682D-02	0.30738D 01	0.49847D 03	0.0
-0.83915D-01	-0.83915D-01	0.93750D 00	0.10137D 03	0.10144D 03	0.0	0.0	0.0
0.69609D 00	0.20909D 01	-0.27769D-01	0.0	0.30823D-01	0.30789D-01	0.30854D-01	0.36000D 02
-0.11755D 05	0.30687D 06	0.77509D 04	0.48887D 03	-0.98939D 02	0.18622D 01	-0.81744D-01	-0.17010D-01
-0.11752D 05	0.30687D 06	0.77510D 04	0.49016D 03	-0.98701D 02	0.14356D 01	0.26957D 02	0.10948D 05
-0.11752D 05	0.23776D 04	0.10000D 05	0.49016D 03	-0.98711D 02	0.0	-0.38999D 01	0.10967D 05
-0.11755D 05	0.23778D 04	0.99999D 04	0.48887D 03	-0.98956D 02	0.42305D 00	0.10144D 03	0.99999D 04
0.31500D 01	-0.16153D 00	0.78856D-01	0.12883D 01	0.24494D 00	-0.42305D 00	0.0	0.0
0.82320D 00	-0.10009D 02	-0.91855D 01	0.10976D-01	-0.13345D 00	0.20571D 01	0.49875D 03	0.0
0.11379D 01	0.11379D 01	-0.91406D 01	0.10139D 03	0.10144D 03	0.0	0.0	0.0
0.22975D 01	-0.28175D 02	-0.95552D-01	0.0	0.30007D-01	0.29973D-01	0.30038D-01	0.37000D 02
-0.11263D 05	0.30678D 06	0.77519D 04	0.48905D 03	-0.99219D 02	0.18061D 01	-0.67882D-01	-0.15798D-01
-0.11262D 05	0.30678D 06	0.77525D 04	0.49010D 03	-0.98990D 02	0.14399D 01	0.26957D 02	0.10952D 05
-0.11262D 05	0.22788D 04	0.10000D 05	0.49010D 03	-0.99001D 02	0.0	-0.39089D 01	0.10971D 05
-0.11263D 05	0.22789D 04	0.99994D 04	0.48905D 03	-0.99235D 02	0.36286D 00	0.10147D 03	0.99994D 04
0.14616D 01	-0.16795D 00	0.57817D 00	0.10536D 01	0.23404D 00	-0.36286D 00	0.0	0.0
-0.93753D 00	0.68799D 01	0.59424D 01	-0.12500D-01	0.91732D-01	0.10423D 01	0.49901D 03	0.0
0.45719D 00	0.45719D 00	0.35156D 00	0.10142D 03	0.10147D 03	0.0	0.0	0.0
-0.25071D 01	0.18612D 02	0.10887D 00	0.0	0.29234D-01	0.29200D-01	0.29264D-01	0.38000D 02
-0.11203D 05	0.30676D 06	0.77530D 04	0.48896D 03	-0.99297D 02	0.19084D 01	-0.71183D-01	-0.17762D-01
-0.11201D 05	0.30676D 06	0.77526D 04	0.49009D 03	-0.99022D 02	0.14403D 01	0.26958D 02	0.10951D 05
-0.11201D 05	0.22664D 04	0.10000D 05	0.49009D 03	-0.99032D 02	0.0	-0.39131D 01	0.10970D 05
-0.11203D 05	0.22661D 04	0.10000D 05	0.48896D 03	-0.99314D 02	0.46400D 00	0.10149D 03	0.10000D 05
0.23122D 01	0.29935D 00	-0.39096D 00	0.11376D 01	0.28230D 00	-0.46400D 00	0.0	0.0
-0.19825D 01	-0.83597D 01	-0.10342D 02	-0.26433D-01	-0.11146D 00	0.92056D 00	0.49854D 03	0.0
0.45079D 00	0.45079D 00	0.35156D 00	0.10142D 03	0.10148D 03	0.0	0.0	0.0
-0.52731D 01	-0.22128D 02	0.23019D 00	0.0	0.29140D-01	0.29106D-01	0.29170D-01	0.38125D 02
-0.11141D 05	0.30675D 06	0.77505D 04	0.48897D 03	-0.99301D 02	0.16280D 01	-0.70084D-01	-0.16480D-01
-0.11139D 05	0.30675D 06	0.77528D 04	0.49009D 03	-0.99052D 02	0.14408D 01	0.26954D 02	0.10950D 05
-0.11139D 05	0.22540D 04	0.10000D 05	0.49009D 03	-0.99063D 02	0.0	-0.39114D 01	0.10969D 05
-0.11141D 05	0.22540D 04	0.99977D 04	0.48897D 03	-0.99314D 02	0.18360D 00	0.10149D 03	0.99977D 04
0.22141D 01	0.14758D-01	0.22753D 01	0.11213D 01	0.25103D 00	-0.18360D 00	0.0	0.0
-0.18370D 01	0.11637D 01	-0.67329D 00	-0.24494D-01	0.15517D-01	0.79750D 00	0.49855D 03	0.0
0.44478D 00	0.44478D 00	0.35156D 00	0.10143D 03	0.10148D 03	0.0	0.0	0.0
-0.48594D 01	0.33092D 01	0.21330D 00	0.0	0.29046D-01	0.29013D-01	0.29076D-01	0.38250D 02
-0.11082D 05	0.30674D 06	0.77491D 04	0.48878D 03	-0.99333D 02	0.14579D 01	-0.77665D-01	-0.16498D-01
-0.11078D 05	0.30674D 06	0.77530D 04	0.49008D 03	-0.99083D 02	0.14412D 01	0.26952D 02	0.10946D 05
-0.11078D 05	0.22416D 04	0.10000D 05	0.49008D 03	-0.99093D 02	0.0	-0.39130D 01	0.10965D 05
-0.11082D 05	0.22416D 04	0.99961D 04	0.48878D 03	-0.99344D 02	0.13073D-01	0.10149D 03	0.99961D 04
0.40301D 01	0.15507D-01	0.39171D 01	0.13047D 01	0.25061D 00	-0.13073D-01	0.0	0.0
-0.25407D 01	-0.56295D 01	-0.81702D 01	-0.33876D-01	-0.75060D-01	0.68483D 00	0.49877D 03	0.0
0.43913D 00	0.43913D 00	0.35156D 00	0.10143D 03	0.10149D 03	0.0	0.0	0.0
-0.66850D 01	-0.14604D 02	0.29490D 00	0.0	0.28954D-01	0.28921D-01	0.28983D-01	0.38375D 02

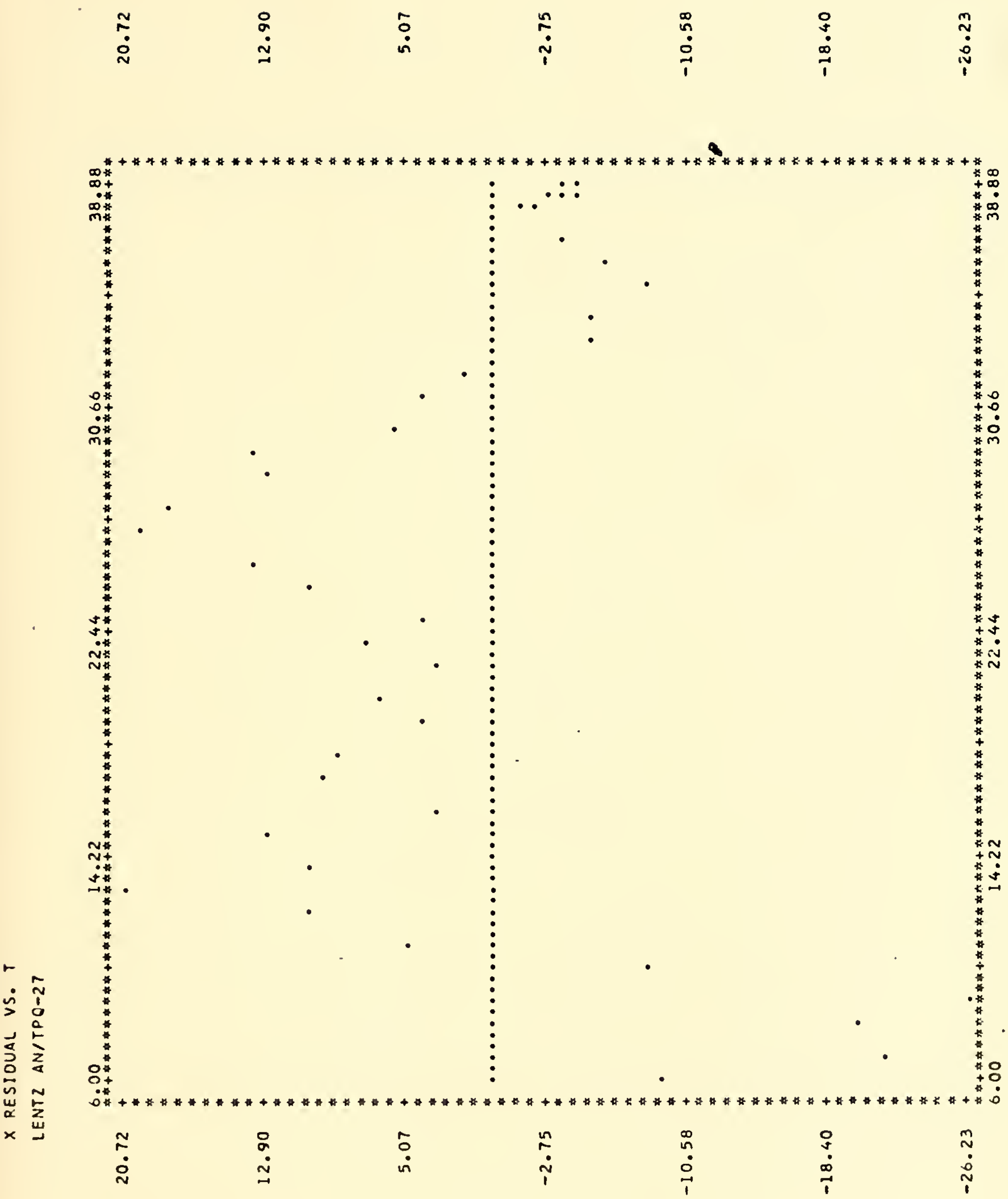
OUTPUT FORMAT

X1	Y1	Z1	XD1	YD1	ZD1	XDD1	YDD1
X2	Y2	Z2	XD2	YD2	ZD2	TF	RA
X3	Y3	Z3	XD3	YD3	ZD3	XGC	YGC
X5	Y5	Z5	XD5	YD5	ZD5	AA	HAT
X6	Y6	Z6	XD6	YD6	ZD6	YNP	
PHD1	PHD2	PHD	HDE	HDEDOT	TG	VEH	THN
PH	PHI	PHC	PS	PSI			
YF	OXE	CTV		G(1,1)	G(4,2)	G(7,3)	T
-0.11020D 05	0.30673D 06	0.77489D 04	0.48889D 03	-0.99302D 02	0.14250D 01	-0.72C00D-01	-0.13767D-01
-0.11017D 05	0.30673D 06	0.77532D 04	0.49008D 03	-0.99113D 02	0.14416D 01	0.26951D 02	0.10948D 05
-0.11017D 05	0.22292D 04	0.10000D 05	0.49008D 03	-0.99123D 02	0.0	-0.39108D 01	0.10967D 05
-0.11020D 05	0.22298D 04	0.99958D 04	0.48889D 03	-0.99313D 02	-0.19405D-01	0.10148D 03	0.99958D 04
0.29498D 01	-0.55812D 00	0.42494D 01	0.11850D 01	0.18936D 00	0.19405D-01	0.0	0.0
-0.17769D 01	0.61109D 01	0.43341D 01	-0.23691D-01	0.81479D-01	0.55368D 00	0.45888D 03	0.0
0.43382D 00	0.43382D 00	0.35156D 00	0.10143D 03	0.10148D 03	0.0	0.0	0.0
-0.46489D 01	0.16289D 02	0.20628D 00	0.0	0.28862D-01	0.28828D-01	0.28691D-01	0.38500D 02
-0.10959D 05	0.30671D 06	0.77495D 04	0.48881D 03	-0.99395D 02	0.14704D 01	-0.75083D-01	-0.16412D-01
-0.10955D 05	0.30671D 06	0.77534D 04	0.49007D 03	-0.99143D 02	0.14421D 01	0.26952D 02	0.10946D 05
-0.10955D 05	0.22168D 04	0.10000D 05	0.49007D 03	-0.99153D 02	0.0	-0.39152D 01	0.10966D 05
-0.10959D 05	0.22168D 04	0.99962D 04	0.48881D 03	-0.99406D 02	0.24661D-01	0.10150D 03	0.99962D 04
0.37725D 01	0.60029D-01	0.38463D 01	0.12640D 01	0.25291D 00	-0.24661D-01	0.0	0.0
-0.29941D 01	-0.97378D 01	-0.12732D 02	-0.39921D-01	-0.12984D 00	0.43198D 00	0.45881D 03	0.0
0.42884D 00	0.42884D 00	0.35156D 00	0.10144D 03	0.10149D 03	0.0	0.0	0.0
-0.77905D 01	-0.25133D 02	0.34755D 00	0.0	0.28771D-01	0.28737D-01	0.28300D-01	0.38625D 02
-0.10899D 05	0.30670D 06	0.77492D 04	0.48876D 03	-0.99388D 02	0.14258D 01	-0.76587D-01	-0.14765D-01
-0.10894D 05	0.30670D 06	0.77535D 04	0.49006D 03	-0.99172D 02	0.14425D 01	0.26951D 02	0.10945D 05
-0.10894D 05	0.22044D 04	0.10000D 05	0.49006D 03	-0.99182D 02	0.0	-0.39150D 01	0.10965D 05
-0.10899D 05	0.22047D 04	0.99957D 04	0.48876D 03	-0.99398D 02	-0.19837D-01	0.10150D 03	0.99957D 04
0.42719D 01	-0.28857D 00	0.42960D 01	0.13085D 01	0.21588D 00	0.19837D-01	0.0	0.0
-0.29156D 01	0.62795D 00	-0.22876D 01	-0.38874D-01	0.83727D-02	0.31030D 00	0.49876D 03	0.0
0.42416D 00	0.42416D 00	0.35156D 00	0.10144D 03	0.10149D 03	0.0	0.0	0.0
-0.75443D 01	0.19701D 01	0.33840D 00	0.0	0.28680D-01	0.28646D-01	0.28709D-01	0.38750D 02
-0.10836D 05	0.30669D 06	0.77471D 04	0.48884D 03	-0.99350D 02	0.11869D 01	-0.72413D-01	-0.11806D-01
-0.10833D 05	0.30669D 06	0.77537D 04	0.49006D 03	-0.99201D 02	0.14429D 01	0.26948D 02	0.10946D 05
-0.10833D 05	0.21920D 04	0.10000D 05	0.49006D 03	-0.99211D 02	0.0	-0.39114D 01	0.10965D 05
-0.10836D 05	0.21930D 04	0.99934D 04	0.48884D 03	-0.99357D 02	-0.25817D 00	0.10149D 03	0.99934D 04
0.34921D 01	-0.96113D 00	0.66327D 01	0.12208D 01	0.14541D 00	0.25817D 00	0.0	0.0
-0.21044D 01	0.64695D 01	0.43851D 01	-0.28059D-01	0.96526D-01	0.18321D 00	0.49883D 03	0.0
0.41976D 00	0.41976D 00	0.35156D 00	0.10144D 03	0.10149D 03	0.0	0.0	0.0
-0.54144D 01	0.17039D 02	0.24429D 00	0.0	0.28589D-01	0.28555D-01	0.28618D-01	0.38875D 02
-0.10776D 05	0.30668D 06	0.77479D 04	0.48880D 03	-0.99479D 02	0.12593D 01	-0.73211D-01	-0.16028D-01
-0.10772D 05	0.30668D 06	0.77539D 04	0.49005D 03	-0.99230D 02	0.14433D 01	0.26949D 02	0.10946D 05
-0.10772D 05	0.21796D 04	0.10000D 05	0.49005D 03	-0.99240D 02	0.0	-0.39170D 01	0.10965D 05
-0.10776D 05	0.21796D 04	0.99940D 04	0.48880D 03	-0.99487D 02	-0.18768D 00	0.10150D 03	0.99940D 04
0.38229D 01	0.30576D-01	0.59856D 01	0.12484D 01	0.24676D 00	0.18768D 00	0.0	0.0
-0.36572D 01	-0.12422D 02	-0.16080D 02	-0.48763D-01	-0.16563D 00	0.57786D-01	0.45883D 03	0.0
0.41563D 00	0.41563D 00	0.35156D 00	0.10145D 03	0.10150D 03	0.0	0.0	0.0
-0.93564D 01	-0.31537D 02	0.42453D 00	0.0	0.28500D-01	0.28466D-01	0.28529D-01	0.39000D 02
-0.10743D 05	0.30667D 06	0.77540D 04	0.49005D 03	-0.99243D 02	0.14435D 01	-0.73211D-01	-0.16028D-01
-0.10743D 05	0.30667D 06	0.77540D 04	0.49005D 03	-0.99243D 02	0.14435D 01	0.26960D 02	0.10972D 05
-0.10743D 05	0.21739D 04	0.10000D 05	0.49005D 03	-0.99253D 02	0.0	-0.39025D 01	0.10991D 05
-0.10743D 05	0.21739D 04	0.10000D 05	0.49005D 03	-0.99253D 02	0.0	0.10145D 03	0.10000D 05
0.0	0.11369D-10	0.18190D-11	0.0	0.0	0.0	0.0	0.0
-0.36572D 01	-0.12422D 02	-0.16080D 02	0.10005D-01	0.10169D 01	-0.60004D-01	0.50000D 03	0.0
0.41380D 00	0.41563D 00	0.35156D 00	0.10145D 03	0.10150D 03	0.0	0.0	0.0
0.19136D 01	-0.31537D 02	0.42453D 00	0.0	0.28500D-01	0.28466D-01	0.28529D-01	0.39058D 02

XI = 2.97849D 01 YI = -4.08013D 00 RI = 3.00631D 01

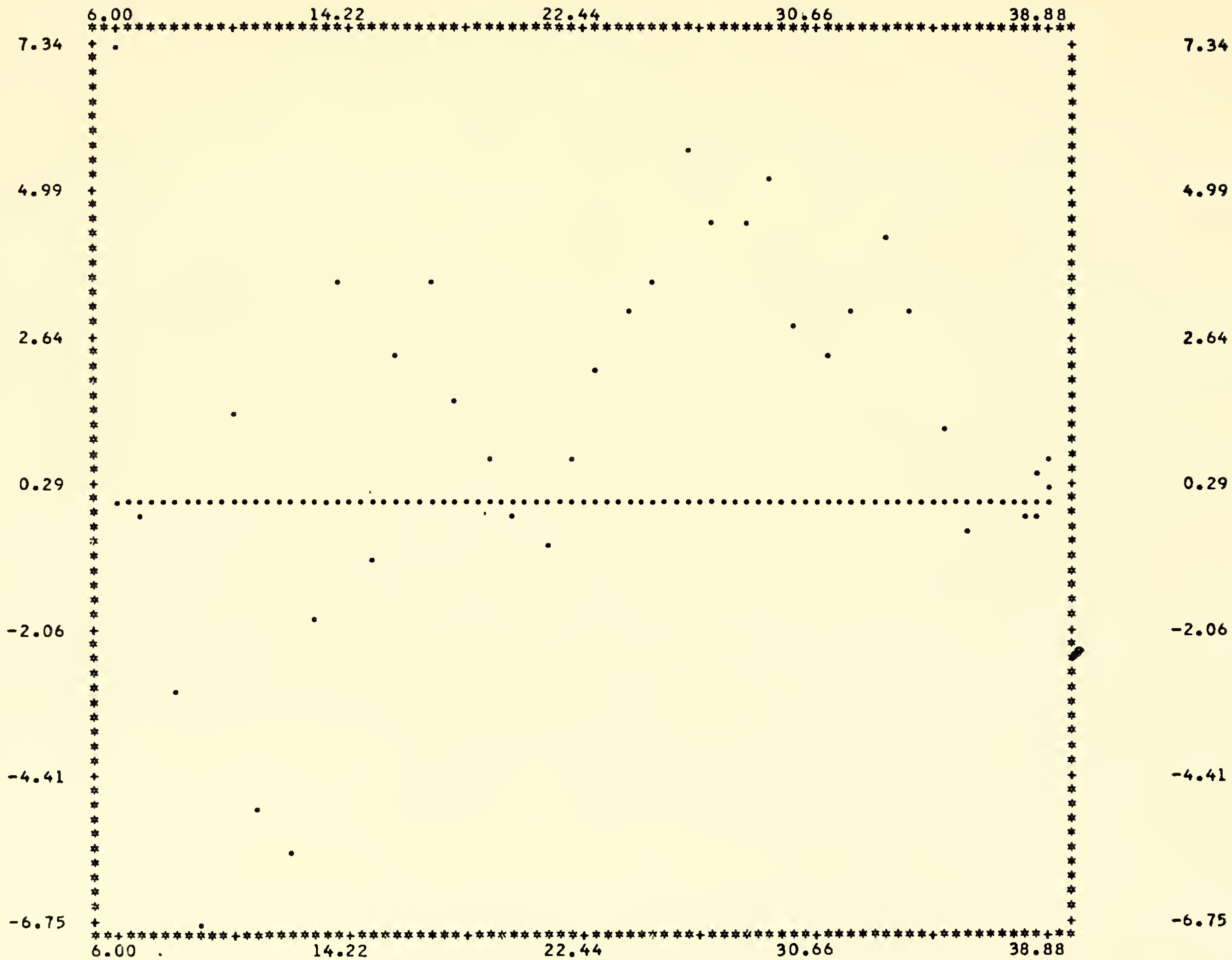
AVG X RESIDUAL = 13.5116 AVG Y RESIDUAL = 4.4865 AVG Z RESIDUAL = 10.6162

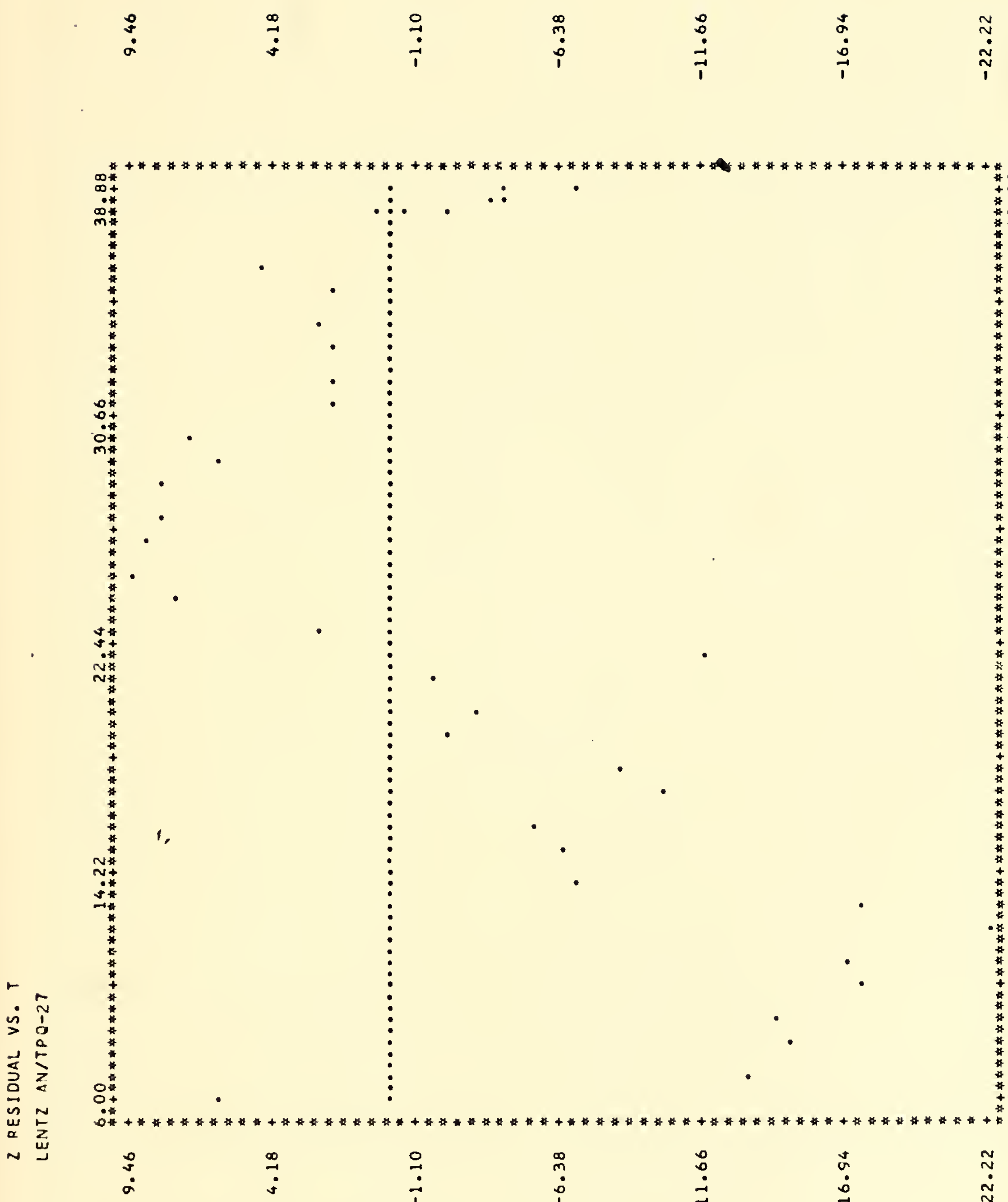
AVG RADIAL RESIDUAL = 17.7594



X RESIDUAL VS. T
LENTZ AN/TPQ-27
6.00
14.22
22.44
30.66
38.88
20.72
12.90
5.07
-2.75
-10.58
-18.40
-26.23

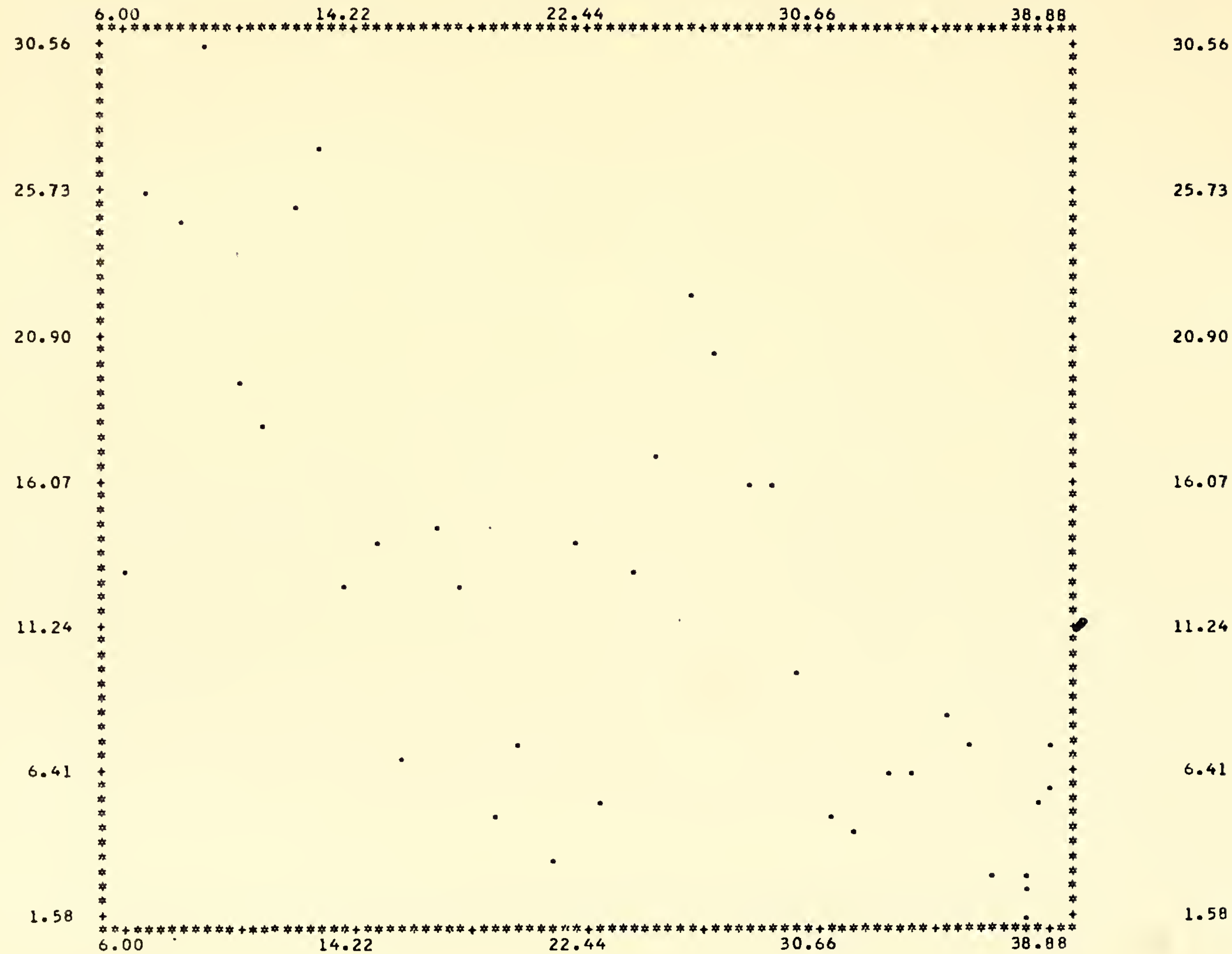
Y RESIDUAL VS. T
LENTZ AN/TPQ-27




Z RESIDUAL VS. T
LENTZ AN/TPQ-27
9.46
4.18
-1.10
-6.38
-11.66
-16.94
-22.22
6.00
14.22
22.44
30.66
38.88

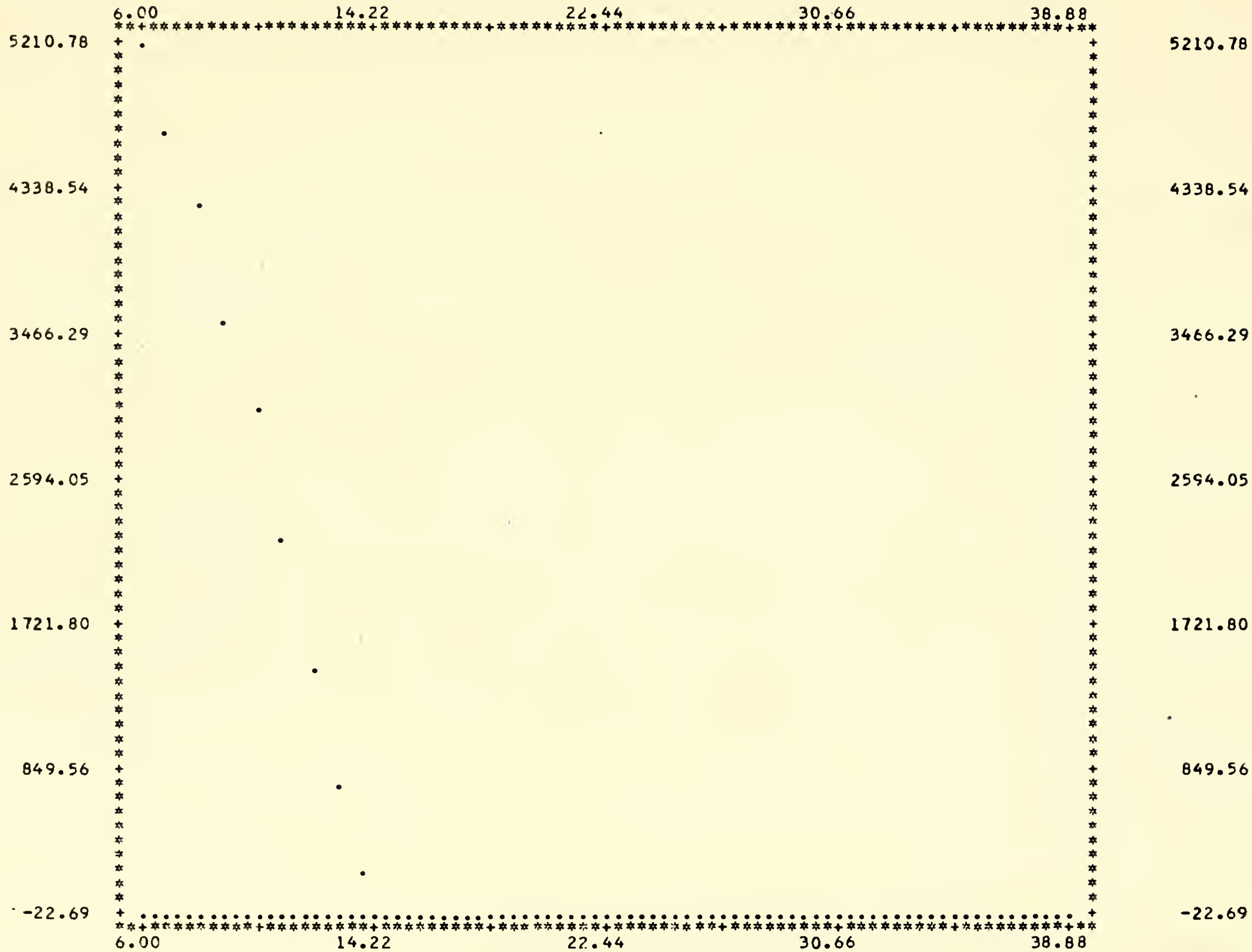
RADIAL RESIDUAL VS. T

LENTZ AN/TPQ-27



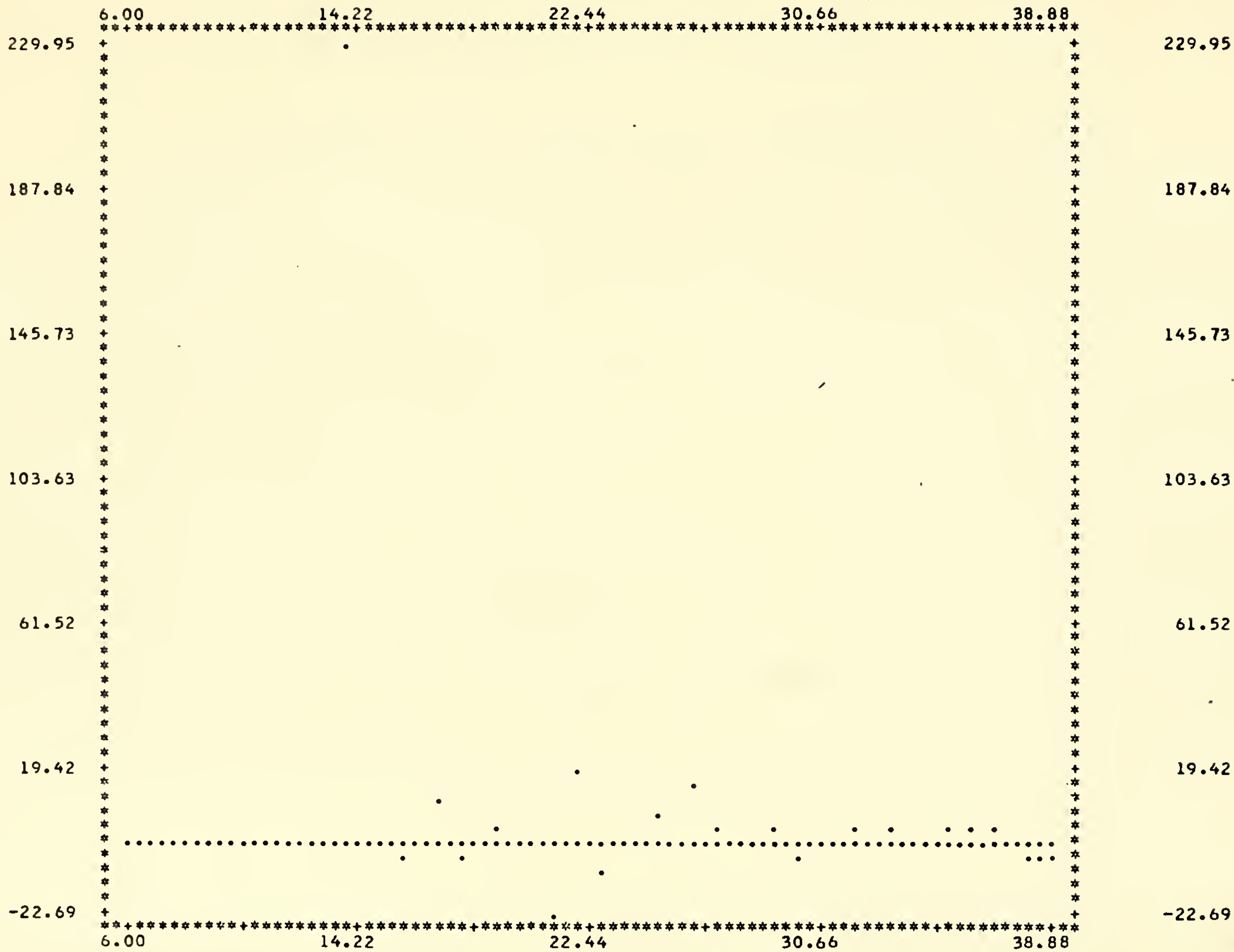
LATERAL ERROR (XE) VS. T

LENTZ AN/TPQ-27



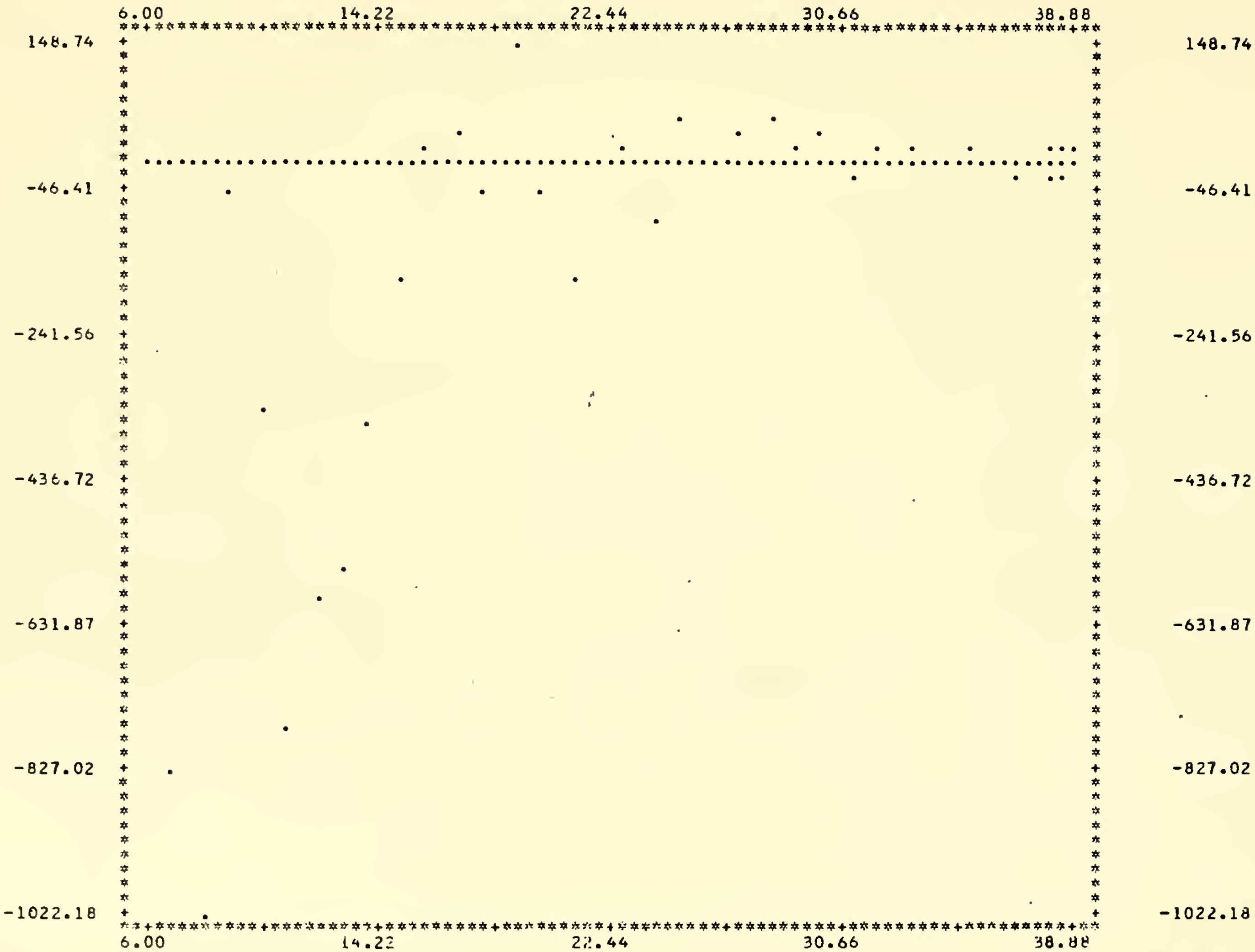
LATERAL ERROR (EXPANDED SCALE) VS. T

LENTZ AN/TPQ-27



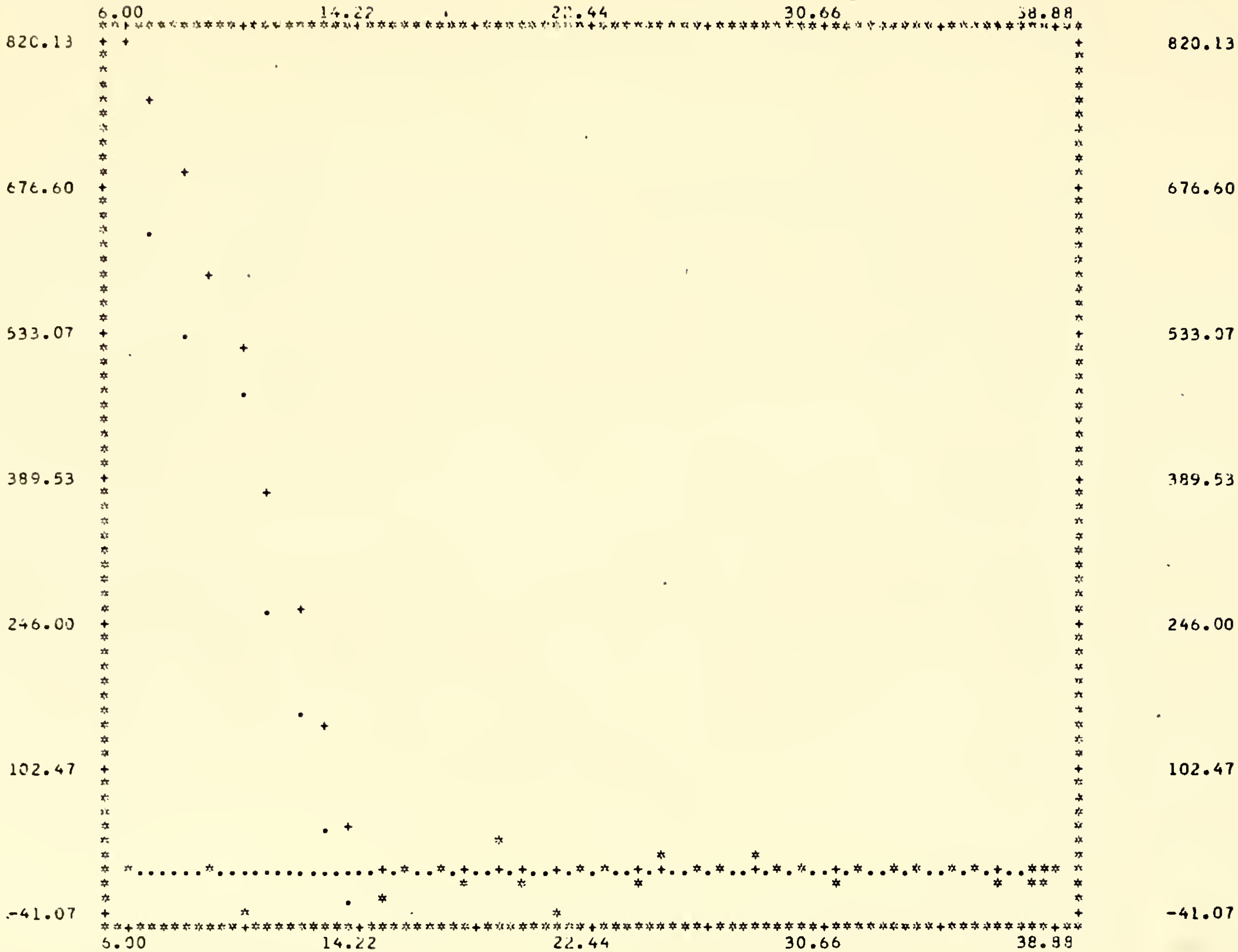
LATERAL ERROR RATE (DXE) VS. T

LENTZ AN/TPQ-27

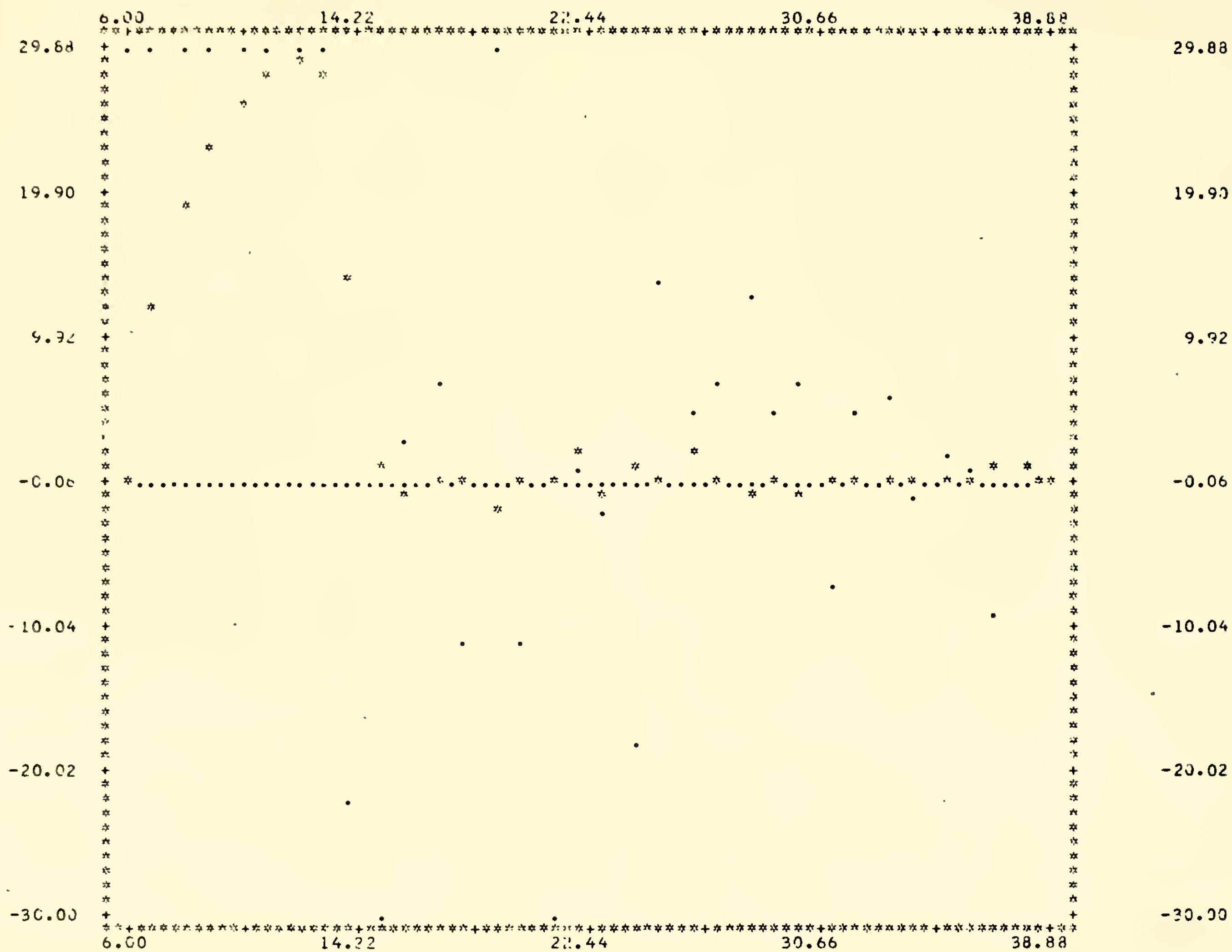


PHD,PHD1, AND PHD2 VS. T

LENTZ AN/TPQ-27

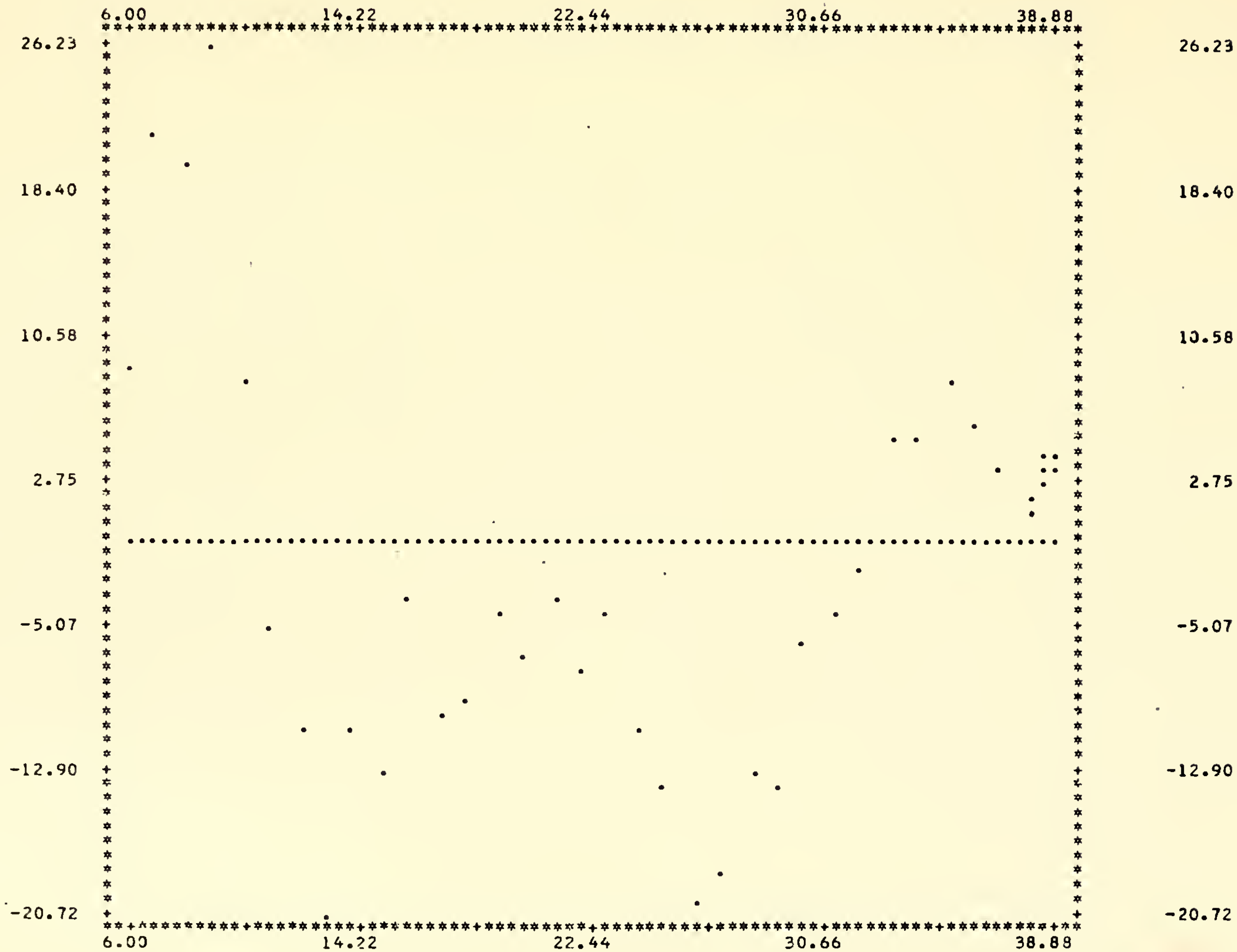


ROLL ANGLE(COMMAND,ACTUAL,BEST) VS. T
LENTZ AN/TPQ-27

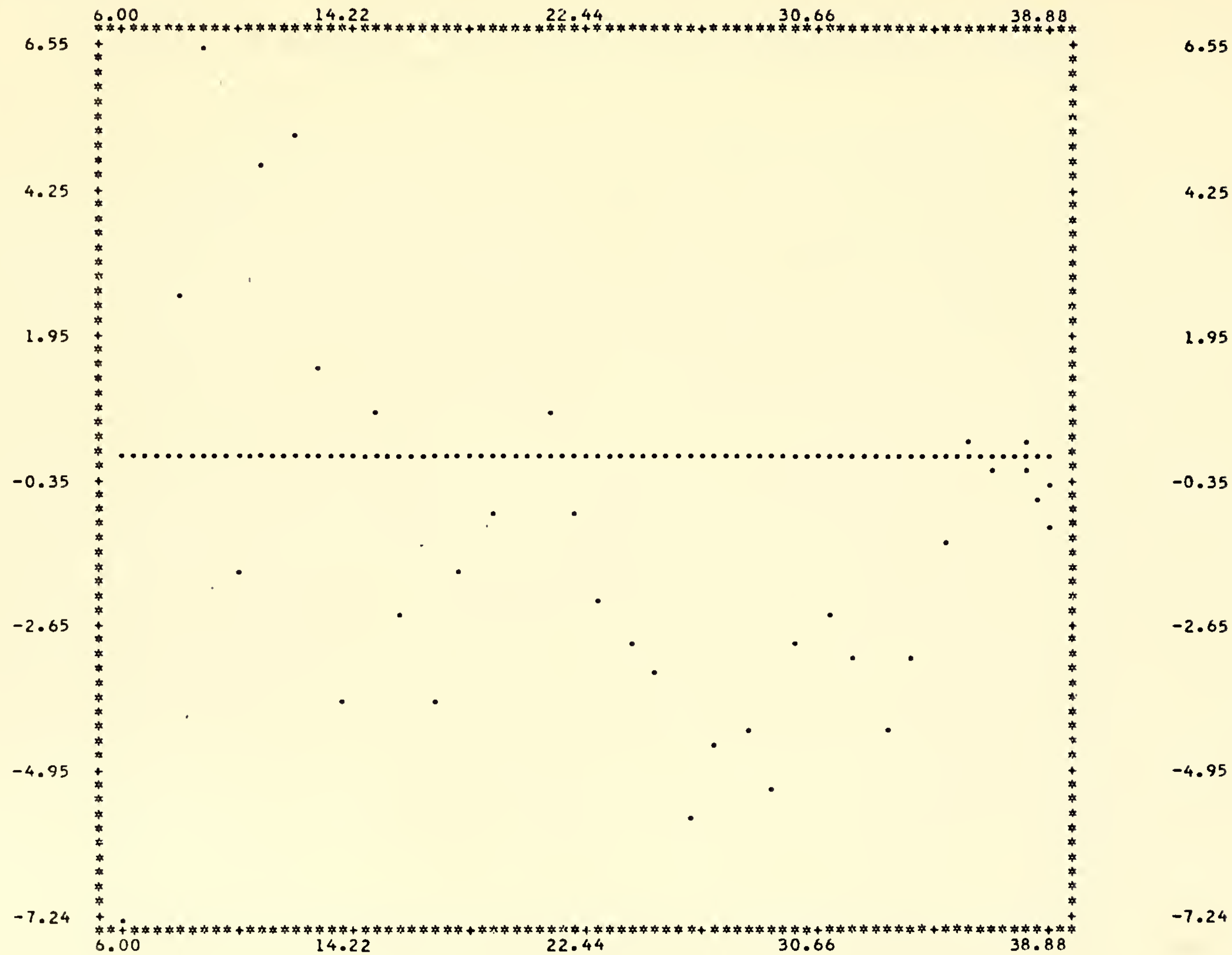


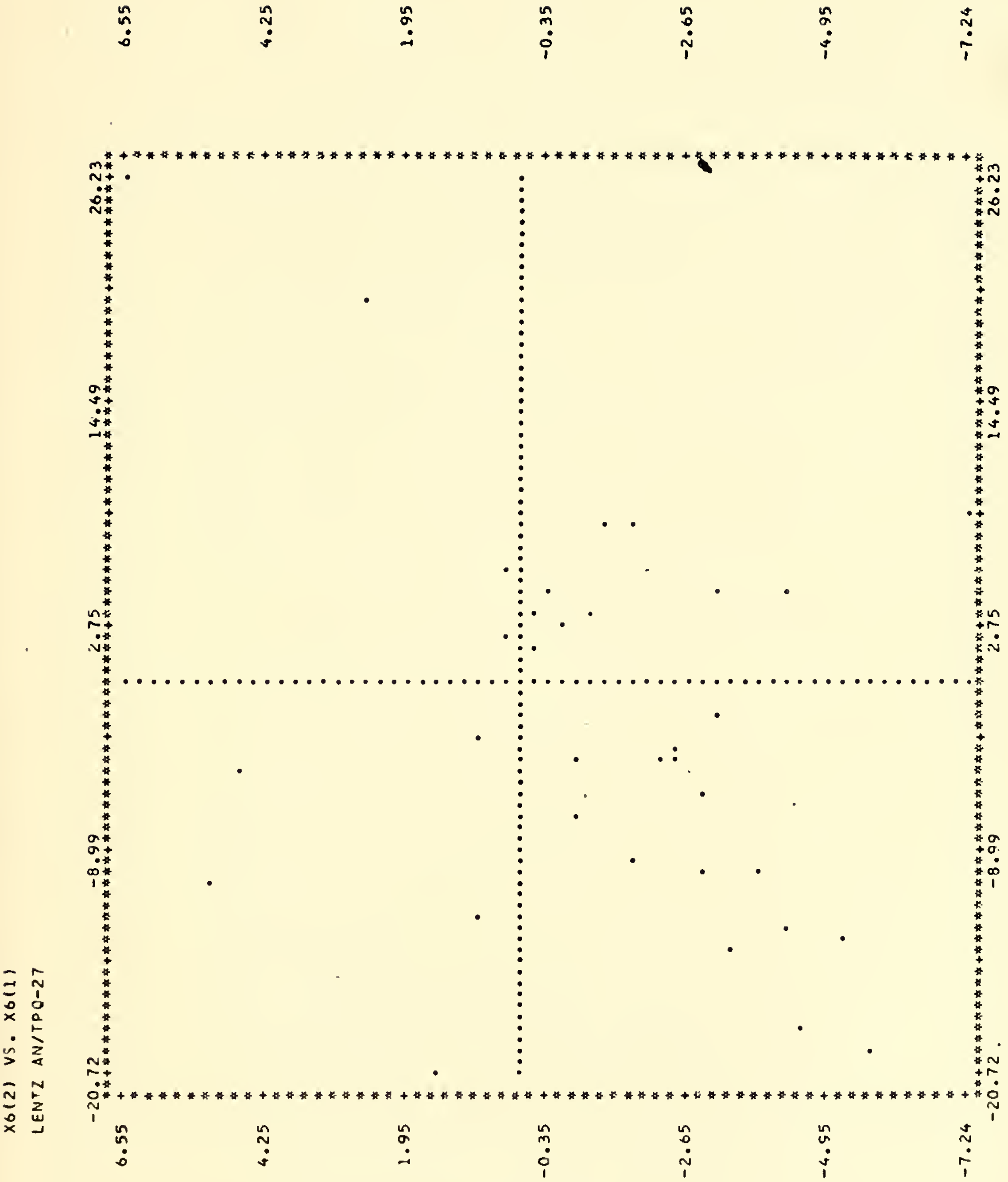
X6(1) VS. T

LENTZ AN/TPQ-27



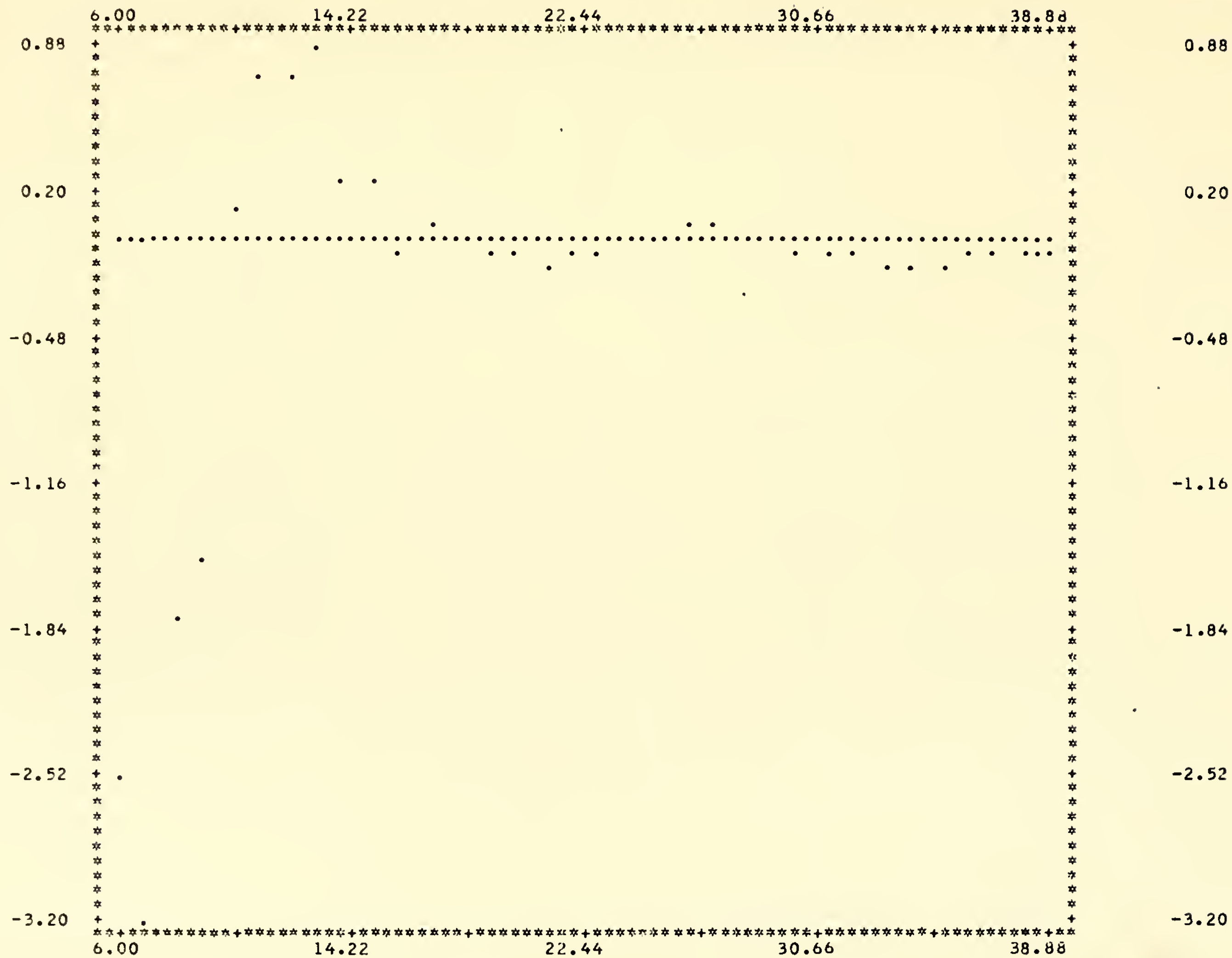
X6(2) VS. T
LENTZ AN/TPQ-27




X6(2) VS. X6(1)
LENTZ AN/TPQ-27
6.55
4.25
1.95
-0.35
-2.65
-4.95
-7.24
-20.72
-8.99
2.75
14.49
26.23

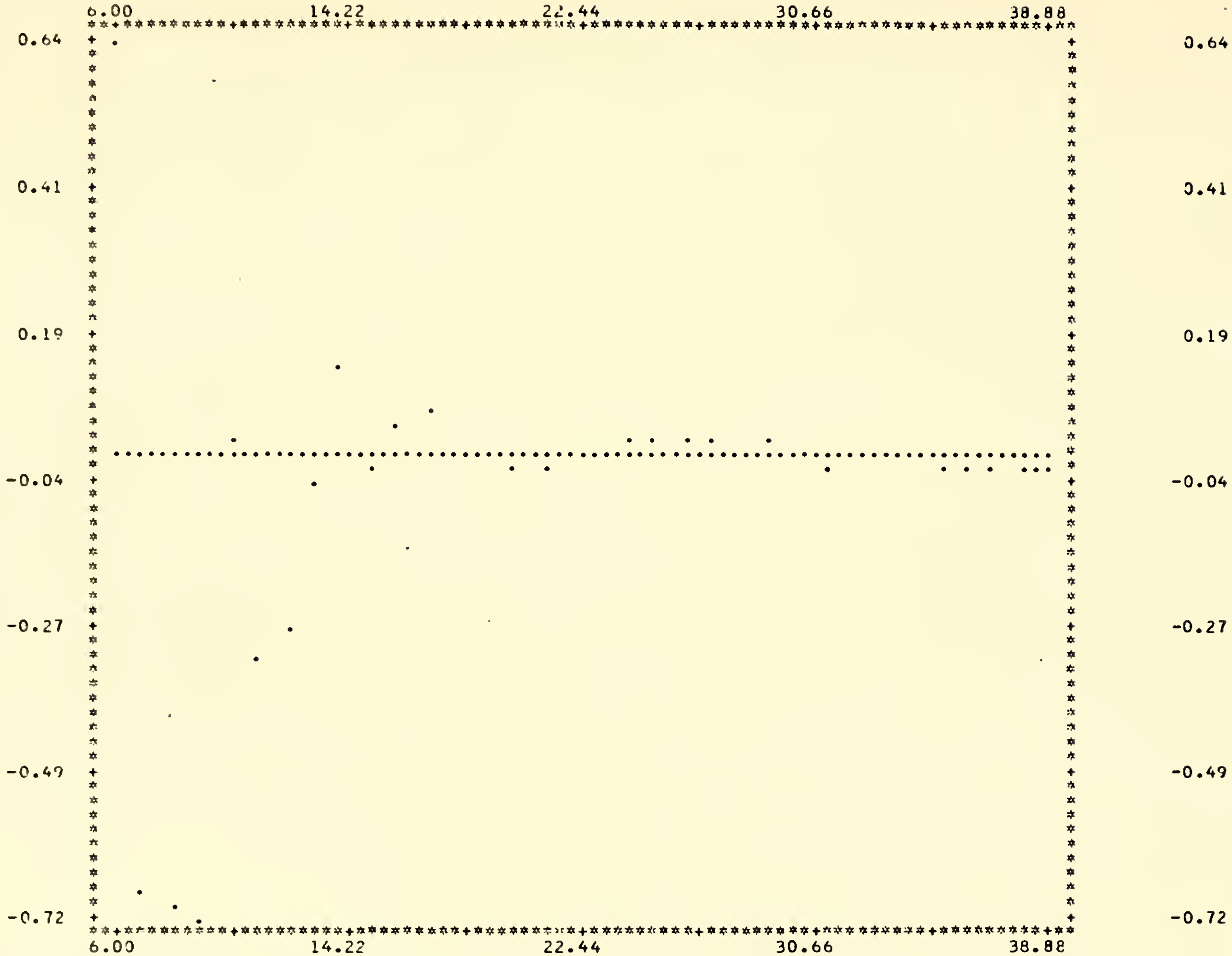
X ACCELERATION ESTIMATE(XDD1(1)) VS. T

LENTZ AN/TPQ-27



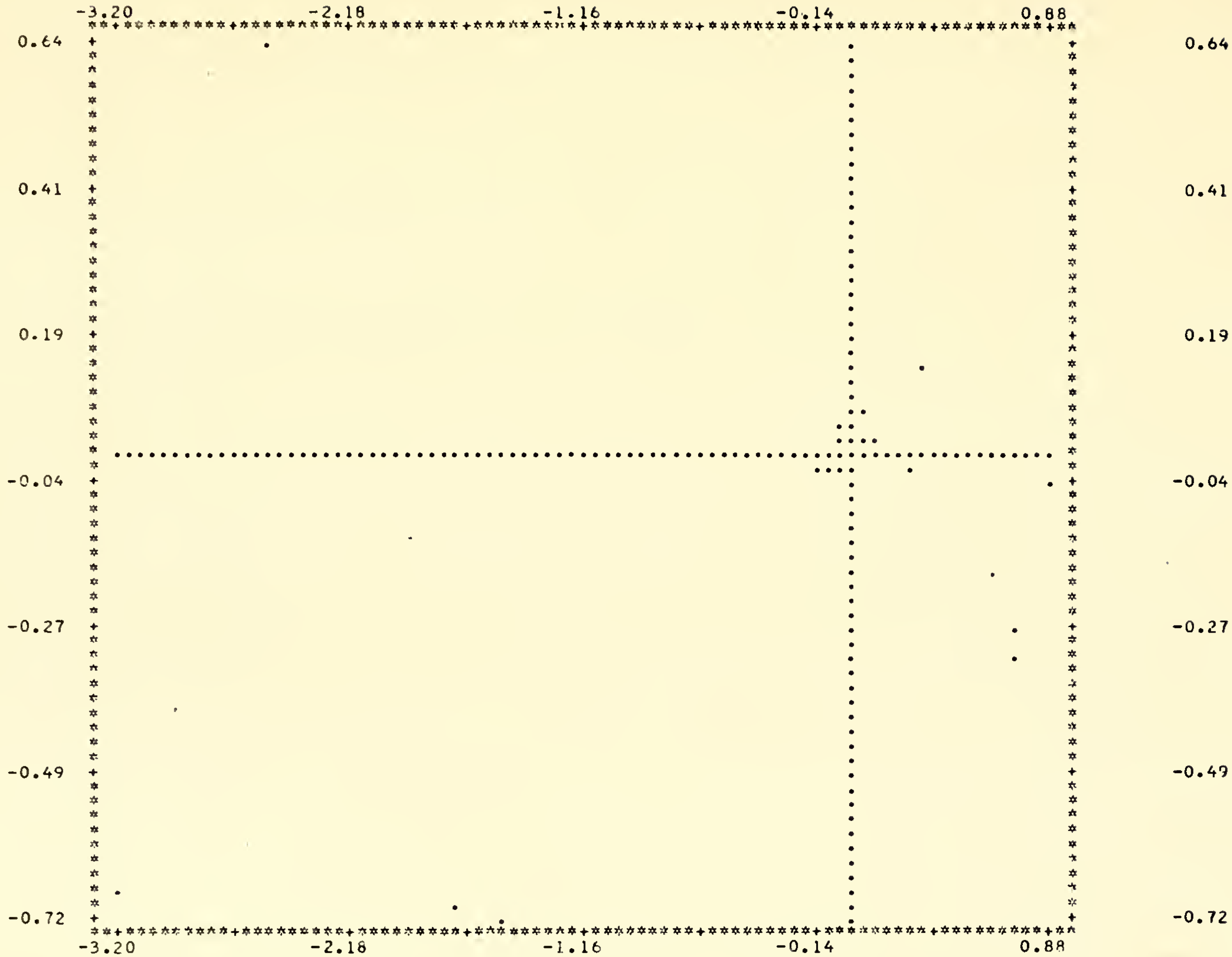
Y ACCELERATION ESTIMATE(XOD1(2)) VS. T

LENTZ AN/TPQ-27



X ACCELERATION VS. Y ACCELERATION

LENTZ AN/TPQ-27



```
                              AN/TPQ-27 SIMULATION

                     COARSE GUIDANCE MODE WITH KALMAN FILTERING

INITIAL CONDITIONS :

   TRUE WIND =  50.00 FT/SEC AT  45.00 DEGREES
   TRUE WIND COMPONENTS =   35.36 35.36
   ESTIMATED WIND =  50.00 FT/SEC AT  45.00 DEGREES
   ESTIMATED WIND COMPONENTS =  35.36 35.36

   RADAR DATA

     MEASUREMENT SIGMAS(R(FT.),AZ(DEG),EL(DEG)) =      9.000000 02     4.000000-01     4.000000-01
     MEASUREMENT BIASES(          "           ) =      0.0             0.0             0.0
     INITIAL VELOCITY MEASUREMENT VALUES =             0.0             0.0             0.0
     RANDOM FORCING ASSUMPTION VALUES (SIGW) =         0.0             0.0             0.0
     SAMPLING INTERVAL FOR RADAR (DTRAD) =   6.000

   AIRCRAFT DATA

     TRUE INITIAL DISPLACEMENT FROM STARTING POINT (NM) =    2.000   2.000   0.0
     TRUE INITIAL GROUND VELOCITY =   550.00 FT/SEC AT  45.00 DEGREES
     TRUE INITIAL GROUND VELOCITY COMPONENTS =   388.91  388.91
     TRUE INITIAL AIR VELOCITY =   500.00 FT/SEC AT  45.00 DEGREES
     ROLL RESPONSE PARAMETER (TB) = 3.300

   TRACK DATA

     NUMBER OF LEGS (NLEGS) =  3
     POSITION OF FIRST LEG =   50.000 NM AT  45.00 DEGREES
     COMPONENTS OF START POINT =   35.355  35.355

   CONTROL DATA

     CONTROL INTERVAL =  1.000
     MAXIMUM BANK ANGLE =   30.00

     G1 =     4.000
     G2 =     4.000

     MIN HEADING ERROR ANGLE FOR COMMAND CORRECTION =    5.000

LEG NUMBER 1

     LEG START POINT (X,Y) =     35.3553    35.3553
     LEG END POINT (X,Y) =       40.3553    44.0156

     LEG LENGTH (NM) =         10.00
     LEG AZIMUTH (DEG) =       30.00

     DESIRED GROUND SPEED (FT/SEC) =   548.13
     DESIRED AIR HEADING (DEG) =   28.52

     AVG RANGE OF LEG FROM RADAR (NM) =         54.84
     AVG AZIMUTH OF LEG FROM RADAR (DEG) =      43.65

     TLEG = 110.852
```

	PHC	PHI	TLEG1	TTURN1	TC21	TT21	HTG	HEG	HTA	HEA	ETFUE	FEST	PHO1	PHO2
0.0	0.0	0.0	0.0	0.0	0.0	0.0	45.00	0.0	45.00	0.0	0.7321	0.4312	0.0	0.0
1.00	0.0	0.0	0.0	0.0	0.0	0.0	45.00	0.0	45.00	0.0	0.7555	0.4333	0.0	0.0
2.00	0.0	0.0	0.0	0.0	0.0	0.0	45.00	0.0	45.00	0.0	0.7789	0.4355	0.0	0.0
3.00	0.0	0.0	0.0	0.0	0.0	0.0	45.00	0.0	45.00	0.0	0.8023	0.4376	0.0	0.0
4.00	0.0	0.0	0.0	0.0	0.0	0.0	45.00	0.0	45.00	0.0	0.8258	0.4397	0.0	0.0
5.00	0.0	0.0	0.0	0.0	0.0	0.0	45.00	0.0	45.00	0.0	0.8492	0.4419	0.0	0.0
6.00	0.0	0.0	0.0	0.0	0.0	0.0	45.00	37.80	45.00	36.92	0.8726	0.4926	0.0	0.0
7.00	0.0	0.0	0.0	0.0	0.0	0.0	45.00	37.80	45.00	36.92	0.8960	0.5028	0.0	0.0
8.00	0.0	0.0	0.0	0.0	0.0	0.0	45.00	37.80	45.00	36.92	0.9195	0.5131	0.0	0.0
9.00	0.0	0.0	0.0	0.0	0.0	0.0	45.00	37.80	45.00	36.92	0.9429	0.5233	0.0	0.0
10.00	0.0	0.0	0.0	0.0	0.0	0.0	45.00	37.80	45.00	36.92	0.9663	0.5335	0.0	0.0
11.00	0.0	0.0	0.0	0.0	0.0	0.0	45.00	37.80	45.00	36.92	0.9898	0.5438	0.0	0.0
12.00	0.0	0.0	0.0	0.0	0.0	0.0	45.00	37.66	45.00	36.71	1.0132	0.5467	0.0	0.0
13.00	0.0	0.0	100.9142	22.2506	14.8530	18.1395	45.00	37.66	45.00	36.71	1.0366	0.5563	0.0	0.0
14.00	-29.8828	-7.8121	99.5156	22.4009	14.9419	18.2288	44.76	37.36	44.74	36.37	1.0595	0.5657	-50.9282	-50.5282
15.00	-29.8828	-13.5819	97.5015	22.8006	15.1769	18.4647	44.13	36.55	44.04	35.46	1.0825	0.5743	-50.2869	0.6413
16.00	-29.8828	-17.8433	95.1112	23.3903	15.5198	18.8087	43.20	35.38	43.02	34.13	1.1039	0.5818	-47.6342	2.6527
17.00	-29.8828	-20.9907	92.5293	24.1282	15.9473	19.2326	42.06	33.93	41.77	32.50	1.1238	0.5876	-43.4711	4.1631
18.00	-29.8828	-23.3153	71.7919	15.9463	13.7697	17.0509	40.76	41.24	40.34	40.89	1.1417	0.8513	-38.1657	5.3054
19.00	-29.8828	-25.0322	69.4074	16.3974	14.1419	17.4271	39.34	39.92	38.78	39.45	1.1575	0.8691	-79.8191	-41.6535
20.00	-29.8828	-26.3003	67.0591	16.8825	14.5409	17.8260	37.84	38.51	37.12	37.91	1.1710	0.8646	-75.3560	3.9631
21.00	-29.8828	-27.2368	64.7734	17.3954	14.9549	18.2417	36.27	37.05	35.40	36.32	1.1821	0.8973	-71.5236	4.3324
22.00	-29.8828	-27.9286	62.5659	17.9318	15.3813	18.6698	34.65	35.54	33.62	34.67	1.1907	0.9094	-66.8956	4.6280
23.00	-29.8823	-28.4394	60.4443	18.4886	15.8159	19.1068	33.00	34.00	31.81	32.99	1.1968	0.9164	-62.0270	4.8686
24.00	-29.8828	-28.8168	60.8346	20.1649	16.5708	19.8628	31.32	31.39	29.97	30.08	1.2002	0.8835	-56.9569	5.0701
25.00	-29.8828	-29.0955	58.8568	20.8197	17.0392	20.3323	29.62	29.73	28.10	28.27	1.2009	0.8844	-45.2054	11.7516
26.00	-29.8828	-29.3013	56.9715	21.4929	17.5115	20.8055	27.91	28.06	26.22	26.44	1.1990	0.8826	-39.3605	5.8449
27.00	-27.1875	-26.7487	55.1841	22.1753	17.9807	21.2755	26.20	26.41	24.36	24.63	1.1944	0.8781	-33.3610	5.9995
28.00	-21.0938	-26.7475	53.5118	22.8402	18.4236	21.7241	24.58	24.82	22.53	22.90	1.1871	0.8710	-27.2995	6.0615
29.00	-15.4688	-23.7990	51.9636	23.4569	18.8361	22.1321	23.09	23.38	20.95	21.33	1.1774	0.8615	-21.4292	5.8703
30.00	0.0	-17.5774	51.5945	22.5681	18.3492	21.6445	21.88	25.09	19.63	23.21	1.1656	0.9711	-16.0244	5.4048
31.00	-29.8828	-20.7943	50.1836	23.0290	18.6581	21.9539	20.75	22.99	18.39	22.02	1.1521	0.9622	-29.3795	-13.3550
32.00	-21.6797	-21.0258	48.7759	23.5369	18.9935	22.2896	19.51	22.81	17.05	20.72	1.1367	0.9515	-25.5689	3.8106
33.00	-16.8750	-19.9407	47.4133	24.0413	19.3214	22.6179	18.31	21.64	15.73	19.46	1.1195	0.9389	-21.3068	4.2620
34.00	0.0	-14.7277	46.1350	24.4717	19.5972	22.8940	17.29	20.66	14.62	18.39	1.1005	0.9246	-17.0651	4.2418

T	PHO	PHI	TLEG1	TTURN1	TG21	TT21	HTG	HEG	HTA	HEA	ETRUE	EEST	PHO1	PHO2
35.00	0.0	-10.8775	44.9454	24.7934	19.8010	23.0979	16.54	19.94	13.81	17.61	1.0802	0.9089	-13.4821	3.5829
36.00	0.0	-8.0339	43.4188	25.7809	20.3710	23.6694	15.98	17.92	13.20	15.40	1.0590	0.8204	-10.8596	2.6225
37.00	0.0	-5.9326	42.3371	25.9640	20.4835	23.7810	15.57	17.52	12.76	14.97	1.0370	0.8007	1.2609	12.1205
38.00	0.0	-4.2825	41.2798	26.1000	20.5658	23.8634	15.27	17.23	12.43	14.65	1.0145	0.7805	2.8958	1.6349
39.00	0.0	-3.2368	40.2393	26.2007	20.6266	23.9243	15.05	17.01	12.13	14.42	0.9916	0.7599	4.1518	1.2560
40.00	0.0	-2.5906	39.2106	26.2754	20.6715	23.9692	14.83	16.85	12.00	14.24	0.9684	0.7390	5.1308	0.9790
41.00	0.0	-1.7656	38.1902	26.3306	20.7047	24.0024	14.76	16.74	11.87	14.12	0.9451	0.7179	5.9083	0.7775
42.00	0.0	-1.3041	41.6014	27.3408	20.5276	23.8252	14.67	17.51	11.77	14.80	0.9215	0.7513	6.5402	0.6319
43.00	0.0	-0.9632	40.5884	27.3744	20.5469	23.8445	14.60	17.45	11.70	14.73	0.8979	0.7724	-0.3351	-6.8752
44.00	0.0	-0.7114	39.5790	27.3995	20.5611	23.8587	14.55	17.39	11.65	14.67	0.8741	0.7533	-0.0676	0.2674
45.00	0.0	-0.5254	38.5723	27.4177	20.5716	23.8692	14.52	17.36	11.61	14.63	0.8503	0.7342	0.1344	0.2020
46.00	0.0	-0.3880	37.5675	27.4313	20.5793	23.8769	14.49	17.33	11.58	14.60	0.8264	0.7151	0.2884	0.1540
47.00	0.0	-0.2866	36.5640	27.4413	20.5850	23.8827	14.47	17.31	11.56	14.58	0.8026	0.6959	0.2884	0.1540
48.00	0.0	-0.2117	37.8147	28.0422	20.5317	23.8293	14.45	17.58	11.54	14.78	0.7786	0.7190	0.2884	0.1540
49.00	0.0	-0.1563	36.8127	28.0481	20.5349	23.8325	14.44	17.57	11.53	14.77	0.7547	0.7007	0.2884	0.1540
50.00	0.0	-0.1155	35.8112	28.0523	20.5373	23.8349	14.44	17.56	11.52	14.76	0.7308	0.6824	0.2884	0.1540
51.00	0.0	-0.0853	34.8102	28.0555	20.5390	23.8366	14.43	17.55	11.51	14.76	0.7068	0.6641	0.2884	0.1540
52.00	0.0	-0.0630	33.8095	28.0578	20.5402	23.8379	14.43	17.55	11.51	14.75	0.6828	0.6458	0.2884	0.1540
53.00	0.0	-0.0465	32.8090	28.0596	20.5413	23.8389	14.42	17.54	11.51	14.75	0.6589	0.6275	0.2884	0.1540
54.00	0.0	-0.0344	29.4487	28.5495	21.1103	24.4033	14.42	15.47	11.50	12.55	0.6349	0.4501	0.2884	0.1540
55.00	0.0	-0.0254	28.4485	28.5504	21.1108	24.4088	14.42	15.47	11.50	12.55	0.6109	0.4284	0.2884	0.1540
	STARTING TURN T =		56.00	TINTRN =	0.0									
56.00	30.0000	7.8240	28.4485	28.5504	21.1108	24.4088	14.66	15.72	11.77	12.82	0.5871	0.4068	0.2884	0.1540
57.00	30.0000	13.6213	28.4485	28.5504	21.1108	24.4088	15.30	16.38	12.46	13.54	0.5639	0.3858	0.2884	0.1540
58.00	30.0000	17.9031	28.4485	28.5504	21.1108	24.4088	16.24	17.35	13.49	14.60	0.5419	0.3561	0.2884	0.1540
59.00	30.0000	21.0655	28.4485	28.5504	21.1108	24.4088	17.40	18.54	14.74	15.90	0.5215	0.3479	0.2884	0.1540
60.00	30.0000	23.4012	28.4485	28.5504	21.1109	24.4088	18.72	19.90	16.18	17.38	0.5029	0.3316	0.2884	0.1540
61.00	30.0000	25.1263	28.4485	28.5504	21.1108	24.4088	20.15	21.37	17.74	19.00	0.4865	0.3174	0.2884	0.1540
62.00	30.0000	26.4004	28.4485	28.5504	21.1108	24.4088	21.67	22.94	19.40	20.71	0.4723	0.3055	0.2884	0.1540
63.00	30.0000	27.3414	28.4485	28.5504	21.1108	24.4088	23.26	24.57	21.13	22.50	0.4605	0.2950	0.2884	0.1540
64.00	30.0000	28.0364	28.4485	28.5504	21.1108	24.4088	24.89	26.25	22.92	24.34	0.4512	0.2890	0.2884	0.1540
65.00	30.0000	28.5497	28.4485	28.5504	21.1108	24.4088	26.55	27.96	24.74	26.23	0.4445	0.2846	0.2884	0.1540
66.00	30.0000	28.9289	28.4485	28.5504	21.1108	24.4088	28.24	29.70	26.59	28.14	0.4404	0.2828	0.2884	0.1540
67.00	30.0000	29.2089	28.4485	28.5504	21.1108	24.4088	29.95	31.46	28.46	30.07	0.4390	0.2837	0.2884	0.1540
68.00	30.0000	29.4157	28.4485	28.5504	21.1108	24.4088	31.67	33.23	30.35	32.02	0.4402	0.2873	0.2884	0.1540

	PHC	PHI	TLEG1	TTURN1	TQ21	TT21	HTG	HEG	HTA	HEA	ETRUE	EEST	PHD1	PHO2
69.00	30.0000	29.5685	28.4485	28.5504	21.1108	24.4088	33.40	35.01	32.24	33.98	0.4442	0.2936	0.2884	0.1540
70.00	30.0000	29.6813	28.4485	28.5504	21.1108	24.4088	35.13	36.60	34.15	35.95	0.4509	0.3026	0.2884	0.1540
71.00	30.0000	29.7646	28.4485	28.5504	21.1108	24.4088	36.87	38.59	36.06	37.93	0.4604	0.3144	0.2884	0.1540
72.00	30.0000	29.8261	28.4485	28.5504	21.1108	24.4088	38.62	40.39	37.98	39.91	0.4726	0.3288	0.2884	0.1540
73.00	30.0000	29.8716	28.4485	28.5504	21.1108	24.4088	40.36	42.19	39.90	41.90	0.4875	0.3460	0.2884	0.1540
74.00	30.0000	29.9052	28.4485	28.5504	21.1108	24.4088	42.11	43.99	41.82	43.88	0.5051	0.3659	0.2884	0.1540
75.00	30.0000	29.9300	28.4485	28.5504	21.1108	24.4088	43.86	45.79	43.75	45.87	0.5255	0.3885	0.2884	0.1540
76.00	30.0000	29.9483	28.4485	28.5504	21.1108	24.4088	45.61	47.60	45.68	47.86	0.5485	0.4137	0.2884	0.1540
77.00	30.0000	29.9618	28.4485	28.5504	21.1108	24.4088	47.37	49.40	47.60	49.86	0.5742	0.4416	0.2884	0.1540
78.00	30.0000	29.9718	28.4485	28.5504	21.1108	24.4088	49.12	51.21	49.53	51.85	0.6025	0.4721	0.2884	0.1540
79.00	30.0000	29.9792	28.4485	28.5504	21.1108	24.4088	50.87	53.02	51.46	53.84	[illegible]	0.5051	0.2884	0.1540
80.00	30.0000	29.9846	28.4485	28.5504	21.1108	24.4088	52.63	54.63	53.39	55.83	0.5470	0.5407	0.2884	0.1540
81.00	30.0000	29.9886	28.4485	28.5504	21.1108	24.4088	54.39	56.64	55.32	57.83	0.4938	0.5344	0.2884	0.1540
82.00	30.0000	29.9916	28.4485	28.5504	21.1108	24.4088	55.14	58.45	57.25	59.82	0.4423	0.4874	0.2884	0.1540
83.00	30.0000	29.9938	28.4485	28.5504	21.1108	24.4088	[illegible].90	60.26	59.18	61.82	0.3932	0.4428	0.2884	0.1540
84.00	30.0000	29.9954	28.4485	28.5504	21.1108	24.4088	[illegible]	62.07	61.11	63.81	0.3464	0.4005	0.2884	0.1540
85.00	30.0000	29.9966	28.4485	28.5504	21.1108	24.4088	61.42	63.89	63.04	65.81	0.3020	0.3609	0.2884	0.1540
86.00	30.0000	29.9975	28.4485	28.5504	21.1108	24.4088	63.18	65.71	64.97	67.30	0.2601	0.3237	0.2884	0.1540
87.00	30.0000	29.9982	28.4485	28.5504	21.1108	24.4088	64.95	67.53	66.90	69.80	0.2209	0.2891	0.2884	0.1540
88.00	30.0000	29.9986	28.4485	28.5504	21.1108	24.4088	66.71	69.35	68.83	71.79	0.1839	0.2570	0.2884	0.1540
89.00	30.0000	29.9990	28.4485	28.5504	21.1108	24.4088	68.48	71.17	70.76	73.79	0.1497	0.2277	0.2884	0.1540
90.00	30.0000	29.9993	28.4485	28.5504	21.1108	24.4088	70.25	73.00	72.69	75.78	0.1181	0.2010	0.2884	0.1540
91.00	30.0000	29.9995	28.4485	28.5504	21.1108	24.4088	72.02	74.83	74.62	77.78	0.0891	0.1769	0.2884	0.1540
92.00	30.0000	29.9996	28.4485	28.5504	21.1108	24.4088	73.79	76.66	76.55	79.77	0.0628	0.1556	0.2884	0.1540
93.00	30.0000	29.9997	28.4485	28.5504	21.1108	24.4088	75.57	78.50	78.48	81.77	0.0392	0.1370	0.2884	0.1540
94.00	30.0000	29.9998	28.4485	28.5504	21.1108	24.4088	77.35	80.34	80.41	83.76	0.0183	0.1212	0.2884	0.1540
95.00	30.0000	29.9998	28.4485	28.5504	21.1108	24.4088	79.13	82.13	82.34	85.76	0.0002	0.1081	0.2884	0.1540
96.00	30.0000	29.9999	28.4485	28.5504	21.1108	24.4088	80.91	84.02	84.27	87.75	-0.0152	0.0977	0.2884	0.1540
97.00	30.0000	29.9999	28.4485	28.5504	21.1108	24.4088	82.70	85.87	86.20	89.75	-0.0279	0.0902	0.2884	0.1540
98.00	30.0000	29.9999	28.4485	28.5504	21.1108	24.4088	84.49	87.72	88.13	91.74	-0.0378	0.0854	0.2884	0.1540

TURN ENDING T = 99.00 TINTRN = 43.00

LEG NUMBER 2

LEG START POINT (X,Y) = 40.3553 44.0156
LEG END POINT (X,Y) = 50.3553 44.0156

LEG LENGTH (NM) = 10.00
LEG AZIMUTH (DEG) = 90.00

DESIRED GROUND SPEED (FT/SEC) = 534.10
DESIRED AIR HEADING (DEG) = 94.05

AVG RANGE OF LEG FROM RADAR (NM) = 63.20
AVG AZIMUTH OF LEG FROM RADAR (DEG) = 45.86

TLEG = 113.762

T	PHC	PHI	TLEG1	TTURN1	TO21	TT21	HTG	HEG	HTA	HEA	ETRUE	EEST	PHO1	PHO2
99.00	0.0	22.1573	95.8523	21.7079	16.8215	20.1140	86.04	89.32	89.80	93.46	-0.0450	0.0833	0.2884	0.1540
100.00	0.0	16.3648	93.8596	22.2024	17.1507	20.4440	87.18	90.51	91.03	94.73	-0.0502	0.0832	0.2884	0.1540
101.00	0.0	12.0867	92.1754	22.5729	17.3938	20.6876	88.03	91.38	91.94	95.67	-0.0538	0.0846	0.2884	0.1540
102.00	0.0	8.9269	90.6970	22.8496	17.5734	20.8675	88.65	92.03	92.61	96.37	-0.0564	0.0872	0.2884	0.1540
103.00	0.0	6.5932	89.3589	23.0556	17.7061	21.0004	89.11	92.51	93.11	96.88	-0.0581	0.0906	0.2884	0.1540
104.00	0.0	4.8696	88.1180	23.2086	17.8040	21.0985	89.46	92.86	93.47	97.26	-0.0592	0.0945	0.2884	0.1540
	TURN COMPLETE	T = 105.00		TINTRN =	49.00									
105.00	0.0	3.5966	86.9452	23.3222	17.8764	21.1710	89.71	93.12	93.74	97.54	-0.0598	0.0990	0.2884	0.1540
106.00	0.0	2.6563	85.8208	23.4064	17.9298	21.2245	89.89	93.32	93.94	97.74	-0.0601	0.1038	-15.4525	-15.7409
107.00	0.0	1.9619	84.7309	23.4687	17.9693	21.2641	90.03	93.46	94.09	97.89	-0.0601	0.1088	-16.4021	-0.9495
108.00	0.0	1.4490	82.7192	22.0452	17.3560	20.6496	90.13	91.35	94.20	95.53	-0.0600	-0.0322	-17.1614	-0.7593
109.00	0.0	1.0702	81.6678	22.0764	17.3769	20.6706	90.21	91.43	94.28	95.61	-0.0597	-0.0301	-4.3890	12.7724
110.00	0.0	0.7904	80.6306	22.0995	17.3924	20.6862	90.26	91.48	94.34	95.67	-0.0594	-0.0279	-4.7468	-0.3578
111.00	0.0	0.5838	79.6034	22.1166	17.4038	20.6976	90.30	91.52	94.38	95.71	-0.0589	-0.0256	-5.0300	-0.2832
112.00	0.0	0.4312	78.5837	22.1293	17.4123	20.7061	90.34	91.55	94.41	95.74	-0.0584	-0.0233	-5.2586	-0.2286
113.00	0.0	0.3185	77.5693	22.1386	17.4185	20.7123	90.36	91.58	94.44	95.77	-0.0579	-0.0209	-5.4475	-0.1888
114.00	0.0	0.2352	78.2271	21.8555	17.1711	20.4644	90.37	90.66	94.46	94.81	-0.0574	-0.0938	-5.6074	-0.1600
115.00	0.0	0.1737	77.2189	21.8606	17.1745	20.4678	90.39	90.68	94.47	94.83	-0.0568	-0.0928	0.5077	6.1152
116.00	0.0	0.1283	76.2129	21.8643	17.1770	20.4703	90.40	90.69	94.43	94.84	-0.0562	-0.0917	0.4647	-0.0431
117.00	0.0	0.0948	75.2086	21.8671	17.1789	20.4722	90.40	90.69	94.49	94.84	-0.0556	-0.0907	0.4341	-0.0306
118.00	0.0	0.0700	74.2054	21.8691	17.1803	20.4736	90.41	90.70	94.49	94.85	-0.0549	-0.0896	0.4128	-0.0213
119.00	0.0	0.0517	73.2031	21.8706	17.1813	20.4746	90.41	90.70	94.49	94.85	-0.0543	-0.0886	0.3983	-0.0144
120.00	0.0	0.0382	73.9070	21.6439	16.9602	20.2531	90.41	89.88	94.50	94.00	-0.0537	-0.1599	0.3890	-0.0093
121.00	0.0	0.0282	72.9057	21.6447	16.9608	20.2536	90.42	89.88	94.50	94.00	-0.0530	-0.1600	6.2915	5.9025
122.00	0.0	0.0208	71.9048	21.6453	16.9612	20.2541	90.42	89.89	94.50	94.00	-0.0524	-0.1602	6.3710	0.0795
123.00	0.0	0.0154	70.9041	21.6458	16.9615	20.2544	90.42	89.89	94.50	94.00	-0.0518	-0.1604	6.4550	0.0840

T	PHC	PHI	TLEG1	TTURN1	TC21	TT21	HTG	HEG	HTA	HEA	ETRUE	EEST	PHD1	PHD2
124.00	0.0	0.0114	69.9036	21.6461	16.9617	20.2546	90.42	89.89	94.50	94.00	-0.0511	-0.1606	6.5431	0.0881
125.00	0.0	0.0084	68.9032	21.6464	16.9619	20.2548	90.42	89.89	94.50	94.00	-0.0505	-0.1607	6.6349	0.0919
126.00	0.0	0.0062	67.6369	21.3346	16.8272	20.1198	90.42	89.43	94.50	93.48	-0.0498	-0.2014	6.7304	0.0955
127.00	0.0	0.0046	66.6367	21.3347	16.8273	20.1199	90.42	89.43	94.50	93.49	-0.0492	-0.2023	10.3196	3.5891
128.00	0.0	0.0034	65.6365	21.3348	16.8274	20.1200	90.42	89.43	94.50	93.49	-0.0486	-0.2031	10.4771	0.1575
129.00	0.0	0.0025	64.6364	21.3349	16.8274	20.1200	90.42	89.43	94.50	93.49	-0.0479	-0.2040	10.6399	0.1628
130.00	0.0	0.0018	63.6363	21.3349	16.8275	20.1200	90.42	89.43	94.50	93.49	-0.0473	-0.2049	10.8061	0.1682
131.00	0.0	0.0014	62.6363	21.3350	16.8275	20.1201	90.42	89.43	94.51	93.49	-0.0466	-0.2057	10.9820	0.1739
132.00	0.0	0.0010	62.8422	21.2942	16.7298	20.0222	90.42	89.06	94.51	93.11	-0.0460	-0.2455	11.1617	0.1797
133.00	0.0	0.0007	61.8422	21.2942	16.7299	20.0222	90.42	89.06	94.51	93.11	-0.0453	-0.2469	14.4836	3.3220
134.00	0.0	0.0005	60.8422	21.2942	16.7299	20.0222	90.42	89.06	94.51	93.11	-0.0447	-0.2484	14.7260	0.2424
135.00	0.0	0.0004	59.8422	21.2942	16.7299	20.0222	90.42	89.06	94.51	93.11	-0.0440	-0.2498	14.9767	0.2507
136.00	0.0	0.0003	58.8421	21.2942	16.7299	20.0222	90.42	89.06	94.51	93.11	-0.0434	-0.2512	15.2361	0.2594
137.00	0.0	0.0002	57.8421	21.2942	16.7299	20.0222	90.42	89.06	94.51	93.11	-0.0427	-0.2526	15.5046	0.2685
138.00	0.0	0.0002	53.4136	21.3131	16.9656	20.2585	90.42	89.98	94.51	94.02	-0.0421	-0.1495	15.7828	0.2782
139.00	0.0	0.0001	52.4136	21.3131	16.9656	20.2585	90.42	89.98	94.51	94.02	-0.0415	-0.1495	7.4997	-8.2831
140.00	0.0	0.0001	51.4136	21.3131	16.9656	20.2585	90.42	89.98	94.51	94.02	-0.0408	-0.1495	7.6454	0.1457
141.00	0.0	0.0001	50.4136	21.3131	16.9656	20.2585	90.42	89.98	94.51	94.02	-0.0402	-0.1496	7.7969	0.1515
142.00	0.0	0.0000	49.4136	21.3131	16.9656	20.2585	90.42	89.98	94.51	94.02	-0.0395	-0.1496	7.9546	0.1576
143.00	0.0	0.0000	48.4136	21.3131	16.9656	20.2585	90.42	89.98	94.51	94.02	-0.0389	-0.1496	8.1187	0.1641
144.00	0.0	0.0000	46.7969	21.3868	17.0442	20.3373	90.42	90.27	94.51	94.32	-0.0382	-0.1153	8.2897	0.1710
145.00	0.0	0.0000	45.7969	21.3868	17.0442	20.3373	90.42	90.27	94.51	94.32	-0.0376	-0.1149	5.4002	-2.8895
146.00	0.0	0.0000	44.7969	21.3868	17.0442	20.3373	90.42	90.27	94.51	94.32	-0.0369	-0.1145	5.5197	0.1195
147.00	0.0	0.0000	43.7969	21.3868	17.0442	20.3373	90.42	90.27	94.51	94.32	-0.0363	-0.1141	5.6446	0.1249
148.00	0.0	0.0000	42.7969	21.3868	17.0442	20.3373	90.42	90.27	94.51	94.32	-0.0357	-0.1137	5.7753	0.1307
149.00	0.0	0.0000	41.7969	21.3868	17.0442	20.3373	90.42	90.27	94.51	94.32	-0.0350	-0.1133	5.9122	0.1369
150.00	0.0	0.0000	42.0859	21.3384	16.9314	20.2242	90.42	89.84	94.51	93.89	-0.0344	-0.1679	6.0557	0.1435
151.00	0.0	0.0000	41.0859	21.3384	16.9314	20.2242	90.42	89.84	94.51	93.89	-0.0337	-0.1682	11.3585	5.3028
152.00	0.0	0.0000	40.0859	21.3384	16.9314	20.2242	90.42	89.84	94.51	93.89	-0.0331	-0.1684	11.6426	0.2841
153.00	0.0	0.0000	39.0859	21.3384	16.9314	20.2242	90.42	89.84	94.51	93.89	-0.0324	-0.1687	11.9413	0.2987
154.00	0.0	0.0000	38.0859	21.3384	16.9314	20.2242	90.42	89.84	94.51	93.89	-0.0318	-0.1689	12.2557	0.3144
155.00	0.0	0.0000	37.0859	21.3384	16.9314	20.2242	90.42	89.84	94.51	93.89	-0.0311	-0.1692	12.5872	0.3314
156.00	0.0	0.0000	36.4294	21.2590	16.8688	20.1615	90.42	89.61	94.51	93.65	-0.0305	-0.1984	12.9370	0.3498
157.00	0.0	0.0000	35.4294	21.2590	16.8688	20.1615	90.42	89.61	94.51	93.65	-0.0298	-0.1990	16.3521	3.4161
158.00	0.0	0.0000	34.4294	21.2590	16.8688	20.1615	90.42	89.61	94.51	93.65	-0.0292	-0.1996	16.8328	0.4798

T	PHC	PHI	TLEG1	TTURN1	TO21	TT21	HTG	HEG	HTA	HEA	ETRUE	EEST	PHO1	PHO2
159.00	0.0	0.0000	33.4294	21.2590	16.8688	20.1615	90.42	89.61	94.51	93.65	-0.0286	-0.2002	17.3416	0.5087
160.00	0.0	0.0000	32.4294	21.2590	16.8688	20.1615	90.42	89.61	94.51	93.65	-0.0279	-0.2008	17.8819	0.5404
161.00	0.0	0.0000	31.4294	21.2590	16.8688	20.1615	90.42	89.61	94.51	93.65	-0.0273	-0.2014	18.4570	0.5751
162.00	0.0	0.0000	28.0981	21.4655	17.1354	20.4286	90.42	90.61	94.51	94.67	-0.0266	-0.0635	18.4570	0.5751
163.00	0.0	0.0000	27.0981	21.4655	17.1354	20.4286	90.42	90.61	94.51	94.67	-0.0260	-0.0625	18.4570	0.5751
164.00	0.0	0.0000	26.0981	21.4655	17.1354	20.4286	90.42	90.61	94.51	94.67	-0.0253	-0.0616	18.4570	0.5751
165.00	0.0	0.0000	25.0981	21.4655	17.1354	20.4286	90.42	90.61	94.51	94.67	-0.0247	-0.0606	18.4570	0.5751
166.00	0.0	0.0000	24.0981	21.4655	17.1354	20.4286	90.42	90.61	94.51	94.67	-0.0240	-0.0597	18.4570	0.5751
167.00	0.0	0.0000	23.0981	21.4655	17.1354	20.4286	90.42	90.61	94.51	94.67	-0.0234	-0.0587	18.4570	0.5751
168.00	0.0	0.0000	20.9781	21.6161	17.2921	20.5857	90.42	91.19	94.51	95.28	-0.0227	0.0265	18.4570	0.5751
	STARTING TURN T = 169.00			TINTRN = 0.0										
169.00	-30.0000	-7.8427	20.9781	21.6161	17.2921	20.5857	90.17	90.95	94.24	95.02	-0.0222	0.0282	18.4570	0.5751
170.00	-30.0000	-13.6351	20.9781	21.6161	17.2921	20.5857	89.52	90.30	93.54	94.33	-0.0224	0.0292	18.4570	0.5751
171.00	-30.0000	-17.9133	20.9781	21.6161	17.2921	20.5857	88.57	89.36	92.52	93.32	-0.0239	0.0290	18.4570	0.5751
172.00	-30.0000	-21.0730	20.9781	21.6161	17.2921	20.5857	87.39	88.20	91.26	92.07	-0.0269	0.0271	18.4570	0.5751
173.00	-30.0000	-23.4068	20.9781	21.6161	17.2921	20.5857	86.06	86.88	89.62	90.66	-0.0320	0.0233	18.4570	0.5751
174.00	-30.0000	-25.1304	20.9781	21.6161	17.2921	20.5857	84.60	85.44	88.26	89.11	-0.0392	0.0173	18.4570	0.5751
175.00	-30.0000	-26.4034	20.9781	21.6161	17.2921	20.5857	83.06	83.92	86.60	87.48	-0.0487	0.0090	18.4570	0.5751
176.00	-30.0000	-27.3436	20.9781	21.6161	17.2921	20.5857	81.46	82.34	84.67	85.77	-0.0606	-0.0017	18.4570	0.5751
177.00	-30.0000	-28.0381	20.9781	21.6161	17.2921	20.5857	79.81	80.70	83.09	84.01	-0.0750	-0.0149	18.4570	0.5751
178.00	-30.0000	-28.5510	20.9781	21.6161	17.2921	20.5857	78.13	79.04	81.26	82.21	-0.0921	-0.0308	18.4570	0.5751
179.00	-30.0000	-28.9299	20.9781	21.6161	17.2921	20.5857	76.42	77.36	79.41	80.38	-0.1117	-0.0492	18.4570	0.5751
180.00	-30.0000	-29.2096	20.9781	21.6161	17.2921	20.5857	74.70	75.65	77.54	78.54	-0.1339	-0.0703	18.4570	0.5751
181.00	-30.0000	-29.4162	20.9781	21.6161	17.2921	20.5857	72.97	73.94	75.66	76.67	-0.1588	-0.0940	18.4570	0.5751
182.00	-30.0000	-29.5688	20.9781	21.6161	17.2921	20.5857	71.23	72.21	73.76	74.80	-0.1863	-0.1203	18.4570	0.5751
183.00	-30.0000	-29.6815	20.9781	21.6161	17.2921	20.5857	69.48	70.48	71.85	72.92	-0.2165	-0.1493	18.4570	0.5751
184.00	-30.0000	-29.7648	20.9781	21.6161	17.2921	20.5857	67.72	68.75	69.94	71.03	-0.2492	-0.1809	18.4570	0.5751
185.00	-30.0000	-29.8263	20.9781	21.6161	17.2921	20.5857	65.97	67.02	68.02	69.14	-0.2846	-0.2152	18.4570	0.5751
186.00	-30.0000	-29.8717	20.9781	21.6161	17.2921	20.5857	64.21	65.28	66.10	67.24	-0.3225	-0.2519	18.4570	0.5751
187.00	-30.0000	-29.9052	20.9781	21.6161	17.2921	20.5857	62.46	63.54	64.18	65.34	-0.3629	-0.2913	18.4570	0.5751
188.00	-30.0000	-29.9300	20.9781	21.6161	17.2921	20.5857	60.70	61.81	62.25	63.44	-0.3738	-0.3052	18.4570	0.5751
189.00	-30.0000	-29.9483	20.9781	21.6161	17.2921	20.5857	58.94	60.07	60.33	61.54	-0.3290	-0.2563	18.4570	0.5751
190.00	-30.0000	-29.9618	20.9781	21.6161	17.2921	20.5857	57.19	58.33	58.40	59.64	-0.2865	-0.2137	18.4570	0.5751
191.00	-30.0000	-29.9718	20.9781	21.6161	17.2921	20.5857	55.43	56.60	56.47	57.74	-0.2465	-0.1716	18.4570	0.5751
192.00	-30.0000	-29.9792	20.9781	21.6161	17.2921	20.5857	53.68	54.86	54.54	55.83	-0.2089	-0.1319	18.4570	0.5751

T	PHC	PHI	TLEG1	TTURN1	TO21	TT21	HTG	HEG	HTA	HEA	ETRUE	EEST	PHO1	PHO2
193.00	-30.0000	-29.9846	20.9781	21.6161	17.2921	20.5857	51.92	53.13	52.61	53.93	-0.1739	-0.0947	18.4570	0.5751
194.00	-30.0000	-29.9886	20.9781	21.6161	17.2921	20.5857	50.17	51.39	50.68	52.02	-0.1414	-0.0600	18.4570	0.5751
195.00	-30.0000	-29.9916	20.9781	21.6161	17.2921	20.5857	48.41	49.66	48.75	50.12	-0.1115	-0.0279	18.4570	0.5751
196.00	-30.0000	-29.9938	20.9781	21.6161	17.2921	20.5857	46.66	47.92	46.82	48.21	-0.0842	0.0016	18.4570	0.5751
197.00	-30.0000	-29.9954	20.9781	21.6161	17.2921	20.5857	44.90	46.19	44.89	46.31	-0.0596	0.0285	18.4570	0.5751
198.00	-30.0000	-29.9966	20.9781	21.6161	17.2921	20.5857	43.15	44.46	42.96	44.40	-0.0377	0.0527	18.4570	0.5751
199.00	-30.0000	-29.9975	20.9781	21.6161	17.2921	20.5857	41.39	42.72	41.03	42.50	-0.0184	0.0742	18.4570	0.5751
200.00	-30.0000	-29.9982	20.9781	21.6161	17.2921	20.5857	39.64	40.99	39.10	40.59	-0.0019	0.0930	18.4570	0.5751
201.00	-30.0000	-29.9986	20.9781	21.6161	17.2921	20.5857	37.88	39.25	37.17	38.69	0.0119	0.1091	18.4570	0.5751
202.00	-30.0000	-29.9990	20.9781	21.6161	17.2921	20.5857	36.12	37.52	35.24	36.78	0.0229	0.1225	18.4570	0.5751
203.00	-30.0000	-29.9993	20.9781	21.6161	17.2921	20.5857	34.37	35.78	33.31	34.88	0.0311	0.1330	18.4570	0.5751

TURN ENDING T = 204.00 TINTRN = 35.00

LEG NUMBER 3

LEG START POINT (X,Y) = 50.3553 44.0156
LEG END POINT (X,Y) = 55.3553 52.6758

LEG LENGTH (NM) = 10.00
LEG AZIMUTH (DEG) = 30.00

DESIRED GROUND SPEED (FT/SEC) = 548.13
DESIRED AIR HEADING (DEG) = 28.52

AVG RANGE OF LEG FROM RADAR (NM) = 71.63
AVG AZIMUTH OF LEG FROM RADAR (DEG) = 47.55

TLEG = 110.852

T	PHC	PHI	TLEG1	TTURN1	TO21	TT21	HTG	HEG	HTA	HEA	ETRUE	EEST	PHO1	PHO2
204.00	0.0	-22.1568	20.9781	21.6161	17.2921	20.5857	32.85	34.28	31.64	33.23	0.0368	0.1410	18.4570	0.5751
205.00	0.0	-16.3645	20.9781	21.6161	17.2921	20.5857	31.73	33.18	30.41	32.02	0.0403	0.1469	18.4570	0.5751
206.00	0.0	-12.0864	20.9781	21.6161	17.2921	20.5857	30.90	32.36	29.51	31.12	0.0424	0.1513	18.4570	0.5751
207.00	0.0	-8.9267	20.9781	21.6161	17.2921	20.5857	30.29	31.76	28.84	30.46	0.0433	0.1546	18.4570	0.5751
208.00	0.0	-6.5931	20.9781	21.6161	17.2921	20.5857	29.84	31.31	28.34	29.97	0.0434	0.1570	18.4570	0.5751
209.00	0.0	-4.8695	20.9781	21.6161	17.2921	20.5857	29.50	30.98	27.97	29.61	0.0428	0.1583	18.4570	0.5751
210.00	0.0	-3.5965	20.9781	21.6161	17.2921	20.5857	29.25	30.74	27.70	29.34	0.0419	0.1601	18.4570	0.5751

TURN COMPLETE T = 211.00 TINTRN = 42.00

T	PHC	PHI	TLEG1	TTURN1	TO21	TT21	HTG	HEG	HTA	HEA	ETRUE	EEST	PHO1	PHO2
211.00	0.0	-2.6563	20.9781	21.6161	17.2921	20.5857	29.08	30.56	27.50	29.14	0.0405	0.1612	18.4570	0.5751
212.00	0.0	-1.9619	20.9781	21.6161	17.2921	20.5857	28.94	30.42	27.36	29.00	0.0390	0.1619	-6.9740	-25.4310
213.00	0.0	-1.4490	20.9781	21.6161	17.2921	20.5857	28.84	30.32	27.25	28.89	0.0372	0.1625	-6.5221	0.4519
214.00	0.0	-1.0702	20.9781	21.6161	17.2921	20.5857	28.77	30.25	27.17	28.81	0.0353	0.1630	-6.2055	0.3166

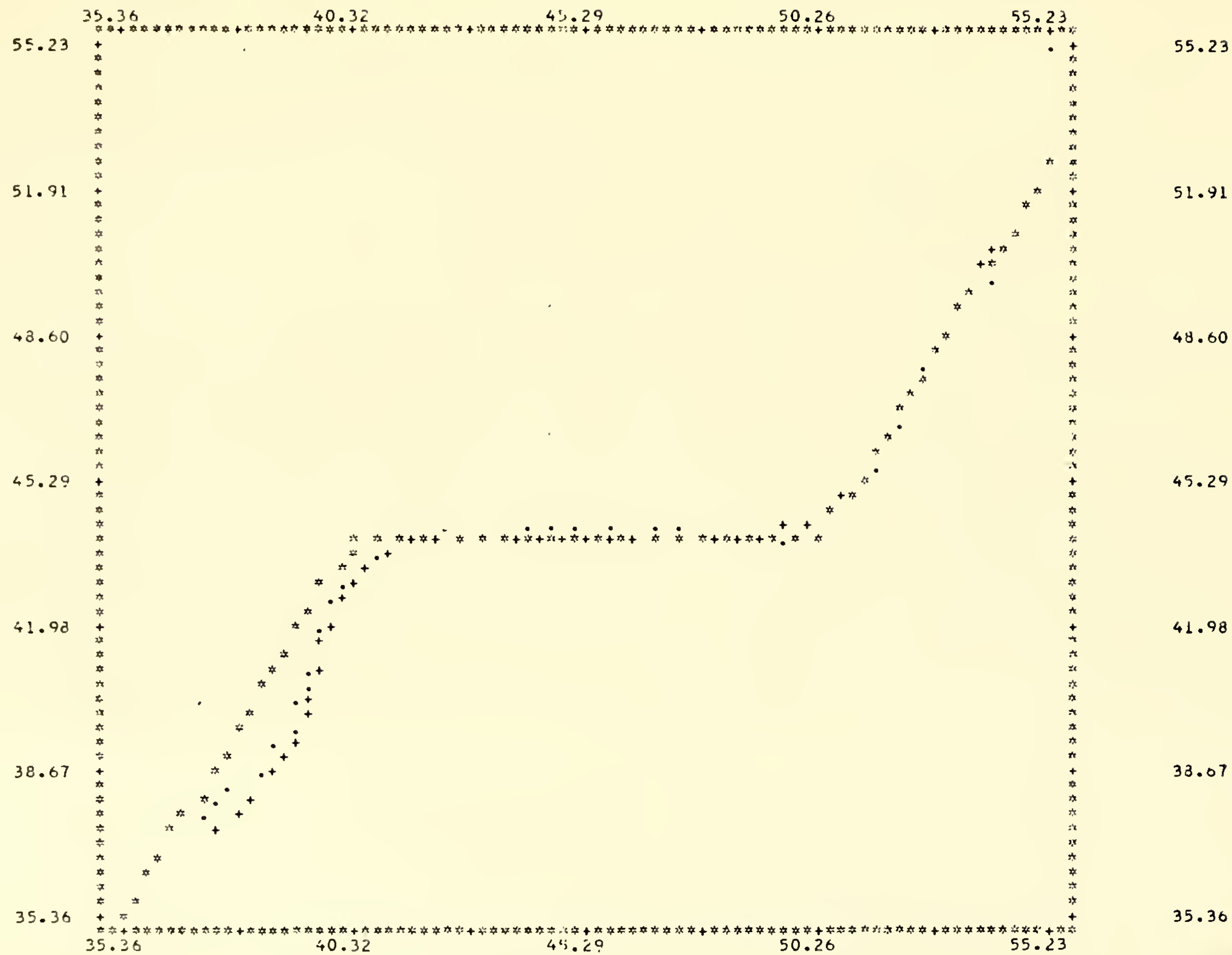
T	PHC	PHI	TLEG1	TTURN1	TO21	TT21	HTG	HEG	HTA	HEA	ETRUE	EEST	PHO1	PHO2
215.00	0.0	-0.7904	20.9781	21.6161	17.2921	20.5857	28.71	30.20	27.11	28.75	0.0334	0.1633	-5.9893	0.2162
216.00	0.0	-0.5838	20.9781	21.6161	17.2921	20.5857	28.67	30.17	27.06	28.71	0.0313	0.1675	-5.8476	0.1417
217.00	0.0	-0.4312	20.9781	21.6161	17.2921	20.5857	28.64	30.14	27.03	28.68	0.0292	0.1677	-5.8958	-0.0482
218.00	0.0	-0.3185	20.9781	21.6161	17.2921	20.5857	28.62	30.12	27.01	28.66	0.0270	0.1679	-5.8510	0.0448
219.00	0.0	-0.2352	20.9781	21.6161	17.2921	20.5857	28.61	30.10	26.99	28.64	0.0249	0.1681	-5.8375	0.0135
220.00	0.0	-0.1737	20.9781	21.6161	17.2921	20.5857	28.60	30.09	26.98	28.63	0.0227	0.1683	-5.8477	-0.0101
221.00	0.0	-0.1283	20.9781	21.6161	17.2921	20.5857	28.59	30.08	26.97	28.62	0.0204	0.1684	-5.9757	-0.0281
222.00	0.0	-0.0948	20.9781	21.6161	17.2921	20.5857	28.58	29.62	26.96	28.11	0.0182	0.0906	-5.9176	-0.0419
223.00	0.0	-0.0700	20.9781	21.6161	17.2921	20.5857	28.57	29.61	26.95	28.11	0.0160	0.0900	-1.5235	4.3942
224.00	0.0	-0.0517	20.9781	21.6161	17.2921	20.5857	28.57	29.61	26.95	28.10	0.0137	0.0894	-1.5250	-0.0015
225.00	0.0	-0.0382	20.9781	21.6161	17.2921	20.5857	28.57	29.61	26.95	28.10	0.0115	0.0888	-1.5318	-0.0068
226.00	0.0	-0.0282	20.9781	21.6161	17.2921	20.5857	28.57	29.60	26.95	28.10	0.0092	0.0882	-1.5426	-0.0109
227.00	0.0	-0.0208	20.9781	21.6161	17.2921	20.5857	28.57	29.60	26.94	28.10	0.0070	0.0875	-1.5567	-0.0141
228.00	0.0	-0.0154	20.9781	21.6161	17.2921	20.5857	28.56	29.20	26.94	27.66	0.0047	0.0148	-1.5733	-0.0166
229.00	0.0	-0.0114	20.9781	21.6161	17.2921	20.5857	28.56	29.20	26.94	27.66	0.0025	0.0135	2.6610	4.2344
230.00	0.0	-0.0084	20.9781	21.6161	17.2921	20.5857	28.56	29.20	26.94	27.66	0.0002	0.0123	2.7037	0.0427
231.00	0.0	-0.0062	20.9781	21.6161	17.2921	20.5857	28.56	29.20	26.94	27.66	-0.0021	0.0110	2.7468	0.0431
232.00	0.0	-0.0046	20.9781	21.6161	17.2921	20.5857	28.56	29.20	26.94	27.65	-0.0043	0.0097	2.7905	0.0438
233.00	0.0	-0.0034	20.9781	21.6161	17.2921	20.5857	28.56	29.20	26.94	27.65	-0.0066	0.0084	2.8352	0.0447
234.00	0.0	-0.0025	20.9781	21.6161	17.2921	20.5857	28.56	29.09	26.94	27.54	-0.0088	-0.0143	2.8809	0.0457
235.00	0.0	-0.0018	20.9781	21.6161	17.2921	20.5857	28.56	29.09	26.94	27.54	-0.0111	-0.0158	4.2371	1.3561
236.00	0.0	-0.0014	20.9781	21.6161	17.2921	20.5857	28.56	29.09	26.94	27.54	-0.0134	-0.0173	4.3071	0.0701
237.00	0.0	-0.0010	20.9781	21.6161	17.2921	20.5857	28.56	29.09	26.94	27.54	-0.0156	-0.0187	4.3794	0.0723
238.00	0.0	-0.0007	20.9781	21.6161	17.2921	20.5857	28.56	29.09	26.94	27.54	-0.0179	-0.0202	4.4541	0.0746
239.00	0.0	-0.0005	20.9781	21.6161	17.2921	20.5857	28.56	29.09	26.94	27.54	-0.0202	-0.0216	4.5312	0.0771
240.00	0.0	-0.0004	20.9781	21.6161	17.2921	20.5857	28.56	28.85	26.94	27.28	-0.0224	-0.0701	4.6110	0.0798
241.00	0.0	-0.0003	20.9781	21.6161	17.2921	20.5857	28.56	28.85	26.94	27.28	-0.0247	-0.0719	7.7633	3.1523
242.00	0.0	-0.0002	20.9781	21.6161	17.2921	20.5857	28.56	28.85	26.94	27.28	-0.0270	-0.0737	7.9061	0.1427
243.00	0.0	-0.0002	20.9781	21.6161	17.2921	20.5857	28.56	28.85	26.94	27.28	-0.0292	-0.0756	8.0541	0.1481
244.00	0.0	-0.0001	20.9781	21.6161	17.2921	20.5857	28.56	28.85	26.94	27.28	-0.0315	-0.0774	8.2078	0.1537
245.00	0.0	-0.0001	20.9781	21.6161	17.2921	20.5857	28.56	28.85	26.94	27.28	-0.0337	-0.0793	8.3675	0.1597
246.00	0.0	-0.0001	20.9781	21.6161	17.2921	20.5857	28.56	29.26	26.94	27.73	-0.0360	0.0026	8.5335	0.1660
247.00	0.0	-0.0000	20.9781	21.6161	17.2921	20.5857	28.56	29.26	26.94	27.73	-0.0383	0.0014	2.8279	-5.70[illegible]
248.00	0.0	-0.0000	20.9781	21.6161	17.2921	20.5657	28.56	29.26	26.94	27.73	-0.0405	0.0002	2.8856	0.[illegible]
249.00	0.0	-0.0000	20.9781	21.6161	17.2921	20.5857	28.56	29.26	26.94	27.73	-0.0428	-0.0010	2.9458	[illegible]

T	PHC	PHI	TLEG1	TTUPN1	TP21	TT21	HTG	HEG	HTA	HEA	ETRUE	EEST	PHD1	PHO2
250.00	0.0	-0.0000	20.9781	21.6161	17.2921	20.5857	28.56	29.26	26.94	27.73	-0.0451	-0.0021	3.0085	0.0627
251.00	0.0	-0.0000	20.9781	21.6161	17.2921	20.5857	28.55	29.26	26.94	27.73	-0.0473	-0.0033	3.0739	0.0654
252.00	0.0	-0.0000	20.9781	21.6161	17.2921	20.5857	28.56	29.13	26.94	27.58	-0.0496	-0.0281	3.1422	0.0683
253.00	0.0	-0.0000	20.9781	21.6161	17.2921	20.5857	28.56	29.13	26.94	27.58	-0.0518	-0.0295	5.0606	1.9184
254.00	0.0	-0.0000	20.9781	21.6161	17.2921	20.5857	28.56	29.13	26.94	27.58	-0.0541	-0.0309	5.1768	0.1162
255.00	0.0	-0.0000	20.9781	21.6161	17.2921	20.5857	28.56	29.13	26.94	27.58	-0.0564	-0.0322	5.2985	0.1217
256.00	0.0	-0.0000	20.9781	21.6161	17.2921	20.5557	28.56	29.13	26.94	27.58	-0.0586	-0.0336	5.4260	0.1275
257.00	0.0	-0.0000	20.9781	21.6161	17.2921	20.5857	28.56	29.13	26.94	27.58	-0.0609	-0.0350	5.5598	0.1338
258.00	0.0	-0.0000	20.9781	21.6161	17.2921	20.5857	28.56	29.37	26.94	27.84	-0.0632	0.0136	5.7004	0.1406
259.00	0.0	-0.0000	20.9781	21.6161	17.2921	20.5857	28.55	29.37	26.94	27.84	-0.0654	0.0126	1.6404	-4.0600
260.00	0.0	-0.0000	20.9781	21.6161	17.2921	20.5857	28.55	29.37	26.94	27.84	-0.0677	0.0116	1.6840	0.0436
261.00	0.0	-0.0000	20.9781	21.6161	17.2921	20.5857	28.56	29.37	26.94	27.84	-0.0700	0.0106	1.7300	0.0459
262.00	0.0	-0.0000	20.9781	21.6161	17.2921	20.5857	28.56	29.37	26.94	27.84	-0.0722	0.0096	1.7785	0.0485
263.00	0.0	-0.0000	20.9781	21.6161	17.2921	20.5857	28.56	29.37	26.94	27.84	-0.0745	0.0086	1.8298	0.0513
264.00	0.0	-0.0000	20.9781	21.6161	17.2921	20.5857	28.56	29.39	26.94	27.86	-0.0767	0.0131	1.3842	0.0544
265.00	0.0	-0.0000	20.9781	21.6161	17.2921	20.5857	28.56	29.39	26.94	27.86	-0.0790	0.0121	1.4328	-0.4514
266.00	0.0	-0.0000	20.9781	21.6161	17.2921	20.5857	28.56	29.39	26.94	27.86	-0.0813	0.0112	1.4778	0.0451
267.00	0.0	-0.0000	20.9781	21.6161	17.2921	20.5857	28.56	29.39	26.94	27.86	-0.0835	0.0102	1.5258	0.0480
268.00	0.0	-0.0000	20.9781	21.6161	17.2921	20.5857	28.56	29.39	26.94	27.86	-0.0858	0.0092	1.5771	0.0512
269.00	0.0	-0.0000	20.9781	21.6161	17.2921	20.5857	28.56	29.39	26.94	27.36	-0.0881	0.0083	1.6318	0.0548
270.00	0.0	-0.0000	20.9781	21.6161	17.2921	20.5857	28.56	29.56	26.94	28.04	-0.0903	0.0462	1.6906	0.0587
271.00	0.0	-0.0000	20.9781	21.6161	17.2921	20.5857	28.56	29.56	26.94	28.04	-0.0926	0.0455	-2.5292	-4.2198
272.00	0.0	-0.0000	20.9781	21.6161	17.2921	20.5857	28.56	29.56	26.94	28.04	-0.0948	0.0447	-2.6261	-0.0969
273.00	0.0	-0.0000	20.9781	21.6161	17.2921	20.5857	28.56	29.56	26.94	28.04	-0.0971	0.0440	-2.7307	-0.1046
274.00	0.0	-0.0000	20.9781	21.6161	17.2921	20.5857	28.56	29.56	26.94	28.04	-0.0994	0.0433	-2.8440	-0.1133

RMS ETRUE = 0.480284
RMS EEST = 0.366266

LENTZ AN/TPQ-27

DESIRED, TRUE, AND ESTIMATED POSITION



```
C
C
C**********AN/TPQ-10 SIMULATION PROGRAM FOR PRECISION GUIDANCE**********
C         MOD 2
C
C
C                  ROBERT E. LENTZ
C
      IMPLICIT REAL*8(A-H,O-Z)
C
      COMMON/AIRCOM/X3(3),XD3(3),WT(3),PHB,TB,DT,PHC,
     1              TG,ITH,IB1,PH,PSD,PS,THC,ANH,ATT,TUP,THN,T
C
      COMMON/RADCOM/SIG(6),BIS(3),X2(3),XD2(3),X1(3),XD1(3),
     1              SIGW(3),WR(3),PS1,PH1,XDD1(3),GNX,GNY,
     2              GNZ,IU,NWLD
C
      COMMON/DERCOM/EKDTAB(112),VKDTAB(112),D2W,HT,IPLC,NNT
C
      COMMON/DIFCOM/HMX,HMN,H,EROR,X,XMX,VDR(16,16),N,NAC,IFL,IFX,ISTF
C
C----DIMENSION ALLOCATION FOR PLOT PACKAGE
      REAL*4 XXA(200),XXB(200),XXC(200),XXD(200),XXE(200),
     1XXF(200),XXG(200)
      REAL*4 YYA(200),YYB(200),YYC(200),YYD(200),YYE(200),
     1YYF(200),YYG(200),YYH(200),YYO(200),YYP(200),YYQ(200),YYR(200),
     2YYS(200),YYT(200),YYU(200),YYS1(200),YYS2(200)
      REAL*8 ATITLE(12),BTITLE(12),CTITLE(12),DTITLE(12),ETITLE(12),
     1FTITLE(12),GTITLE(12),HTITLE(12),OTITLE(12),PTITLE(12),QTITLE(12),
     2RTITLE(12),STITLE(12),TTITLE(12),UTITLE(12),VTITLE(12),WTITLE(12),
     3XTITLE(12),YTITLE(12),ZTITLE(12)
C
      DIMENSION X4(3),X5(3),XD5(3),X6(3),XD6(3),X7(3),WH(3),EM1(3,3),
     1EM2(3,3),EV1(3),NTAB(7),NENT(7),GKD(112,7),XKD(112,7),SENCO(6),
     2FILRES(4),SUMSQR(4),EM3(3,3)
C
      EXTERNAL DER,OUT
C
      DATA DEG,RE,WE/57.295779513082321,20926428.,7.292116D-05/
      DATA G,G2/32.174049,16.087007/
      DATA SENCO,FILRES,SUMSQR/6*0.D0,4*0.D0,4*0.D0/
      DATA X4,X5,XD5,X6,XD6/3*0.D0,3*0.D0,3*0.D0,3*0.D0,3*0.D0/
      DATA X7,WH,EM1,EM2/3*0.D0,3*0.D0,9*0.D0,9*0.D0/
      DATA EM3,EV1/9*0.D0,3*0.D0/
      DATA NTAB,NENT,GKD,XKD/7*0,7*0,784*0.D0,784*0.D0/
      DATA NTBP,ITAB1,NSTP,NSAMPL,JTH,KTH/0,0,12,-1,8,16/
      DATA IMOD,LNC,ILVL,JAVG,IIFLG/7,0,0,0,0/
      DATA YNP,TSTF,TF,RSRA,RA,XGC,YGC/7*0.D0/
```

```
      DATA G1,G2,TLVL,TLVL1,THA,TTHD,TP,ENSTP/75.,75.,2.,3.,0.,          LENT0490
     1      13.5,0.,12./                                                 LENT0500
      DATA CTV,XE,XEO,XEEXPN,DXE,HDE,HDEOLD,HDEDOT,PHDAVG/9*0.D0/        LENT0510
      DATA PHD1,PHD2,PHD/3*0.D0/                                         LENT0520
      DATA ZIP/0.D0/                                                     LENT0530
C                                                                        LENT0540
C                                                                        LENT0550
C----PRIMARY ARRAY DEFINITIONS                                           LENT0560
C     X1: STATE ESTIMATION VECTOR IN RADAR COORDINATES                   LENT0570
C     X2: TRUE COORDINATES OF AIRCRAFT IN RADAR FRAME                    LENT0580
C     X3: TRUE COORDINATES OF AIRCRAFT IN TARGET FRAME                   LENT0590
C     X4: USED FOR TEMPOARAY STORAGE ONLY                                LENT0600
C     X5: ESTIMATED COORDINATES OF THE TARGET IN THE "PRIMED" FRAME      LENT0610
C     X6: ESTIMATED COORDINATES OF TARGET IN THE "DOUBLE PRIMED" SYSTEM  LENT0620
C     X7: ESTIMATED TARGET-AIRCRAFT VECTOR IN RADAR FRAME                LENT0630
C                                                                        LENT0640
C     EV1: TRUE TARGET COORDINATES IN THE RADAR FRAME                    LENT0650
C     WT: TRUE WIND VECTOR IN THE TARGET FRAME                           LENT0660
C     WH: ESTIMATED WIND VECTOR IN THE TARGET FRAME                      LENT0670
C     WR: ESTIMATED WIND VECTOR IN THE RADAR FRAME                       LENT0680
C                                                                        LENT0690
C----NON-ZERO COMMON INITIALIZATIONS                                     LENT0700
      ITH=-1                                                             LENT0710
      TG=1000.0                                                          LENT0720
      N=16                                                               LENT0730
      IFX=1                                                              LENT0740
      ISTF=0                                                             LENT0750
      NAC=4                                                              LENT0760
      EROR=1.D-06                                                        LENT0770
      XMX=3000.D0                                                        LENT0780
      NWLD=-1                                                            LENT0790
      IU=635897                                                          LENT0800
C                                                                        LENT0810
C----READ DATA FOR PROBLEM STARTING WITH POSITION AND RADAR VALUES       LENT0820
      READ(5,2) WT,WH                                                    LENT0830
      READ(5,2) BIS,SIG(4),SIG(5),SIG(6)                                 LENT0840
      READ(5,2) SIG(1),SIG(2),SIG(3)                                     LENT0850
      READ(5,2) SIGW                                                     LENT0860
      READ(5,2) X3                                                       LENT0870
      READ(5,2) XD3                                                      LENT0880
      READ(5,3) PHB,TB,DT,DTCON,PHI,TWLD,MWLD                            LENT0890
C                                                                        LENT0900
C----READ DIVE MODE DATA                                                 LENT0910
      READ(5,2) T3,THNM,THDM,THD,ATT,ATH                                 LENT0920
      READ(5,102) ANH,TUP,HTOL,ATOL,S,IB1,IACC                           LENT0930
C                                                                        LENT0940
C----READ TITLES FOR PLOTTED OUTPUT                                      LENT0950
      READ(5,101) ATITLE,BTITLE,CTITLE,DTITLE,ETITLE,FTITLE,             LENT0960
```

```
     1GTITLE,HTITLE,OTITLE,PTITLE,QTITLE,RTITLE,STITLE,TTITLE,UTITLE,    LENT0970
     2VTITLE,WTITLE,XTITLE,YTITLE,ZTITLE                                 LENT0980
C                                                                        LENT0990
C----READ BOMBING DATA TABLE CONSTANTS AND FIND CORRECT TABLE            LENT1000
      READ(5,102) WBLX,WBLY,BFF,D,W,NTB                                  LENT1010
      READ(5,9)NTABS,(NTAB(I),NENT(I),I=1,NTABS)                         LENT1020
      DO 100 I=1,NTABS                                                   LENT1030
      NNT=NENT(I)                                                        LENT1040
  100 READ(5,1)(XKD(J,I),GKD(J,I),J=1,NNT)                               LENT1050
C                                                                        LENT1060
      IF(NTB.EQ.NTBP)      GO TO 114                                     LENT1070
      NTBP=NTB                                                           LENT1080
      DO 111 I=1,NTABS                                                   LENT1090
      IF(NTB.EQ.NTAB(I))      GO TO 112                                  LENT1100
  111 CONTINUE                                                           LENT1110
      WRITE(6,90)NTB                                                     LENT1120
      GO TO 999                                                          LENT1130
  112 NNT=NENT(I)                                                        LENT1140
      DO 113 J=1,NNT                                                     LENT1150
      EKDTAB(J)=GKD(J,I)                                                 LENT1160
  113 VKDTAB(J)=XKD(J,I)                                                 LENT1170
  114 IPLC=NNT/2                                                         LENT1180
      IDTCON=DTCON/DT+0.5                                                LENT1190
C                                                                        LENT1200
C----TH IS THE ANGLE BETWEEN THE TARGET AND THE RADAR                    LENT1210
      TH=THNM/THDM                                                       LENT1220
      STH=DSIN(TH/DEG)                                                   LENT1230
      CTH=DCOS(TH/DEG)                                                   LENT1240
C                                                                        LENT1250
C----SET UP MATRIX FOR COORDINATE TRANSFORMATION                         LENT1260
      EM1(1,1)=1.DO                                                      LENT1270
      EM1(2,2)= CTH                                                      LENT1280
      EM1(2,3)=STH                                                       LENT1290
      EM1(3,2)=-STH                                                      LENT1300
      EM1(3,3)= CTH                                                      LENT1310
C                                                                        LENT1320
      EV1(1)=0.DO                                                        LENT1330
      EV1(2)=RE*STH                                                      LENT1340
      EV1(3)=-RE+RE*CTH                                                  LENT1350
C                                                                        LENT1360
      DO 11 I=1,3                                                        LENT1370
      DO 11 J=1,3                                                        LENT1380
      EM2(I,J)=EM1(J,I)                                                  LENT1390
      EM3(I,J)=0.DO                                                      LENT1400
   11 IF(I.EQ.J)      EM3(I,J)=1.DO                                      LENT1410
C                                                                        LENT1420
C----PHI IS RADAR LATITUDE                                               LENT1430
      SPH=DSIN(PHI/DEG)                                                  LENT1440
```

```
      CPH=DCOS(PHI/DEG)
C
C----TRANSFORM ESTIMATED WIND VALUES FROM TARGET TO
C    RADAR COORDINATE SYSTEM.
      CALL MATMLT(EM1,WH,WR)
C
C----PRINT OUT INITAL VALUES AND CONSTANTS FOR THE PROBLEM
      WRITE(6,5) WT,WH,MWLD,TWLD,SIG(1),SIG(2),SIG(3),BIS,
     1          SIG(4),SIG(5),SIG(6),SIGW,DT,X3,XD3
      WRITE(6,60) TB,PHB
      WRITE(6,61) DTCON,G1,G2
      WRITE(6,62) WBLX,WBLY,BFF,D,W,NTB
      WRITE(6,63) THD,ATT,ATH,ANH,TUP,HTOL,ATOL,THNM,
     1          THDM,IB1,IACC
C
C----DIVE EQUATION CONSTANTS
C    IB1=0 WHEN IN LEVEL BOMBING MODE
      IF(THD.GT.0.D0)      IB1=1
      D2W=D*D/(144.D0*W)*BFF
      CD4=THD/DEG
      CD1=DEXP(ATH*CD4/ANH)
      CD2=(CD1-1.D0)/ATH
      CTHD=DCOS(CD4)
      STHD=DSIN(CD4)
      TTHD=STHD/CTHD
      CD4=(ATH+ATH)/(ANH*ANH+4.D0*ATH*ATH)
      CD3=CD4*(CD1*CD1*(CTHD+.5D0*ANH/ATH*STHD)-1.D0)
      CD4=CD4*(CD1*CD1*(STHD-.5D0*ANH/ATH*CTHD)+.5D0*ANH/ATH)
      ATTL=ATOL/(DEG*CTHD*CTHD)
C
C***********************************************************************
C
C----BEGIN MAJOR LOOP
C
   12 ITH=ITH+1
C
      CALL ARCRFT
C
C----TRANSFORM FROM TARGET TO RADAR COORDINATE SYSTEM
      CALL MATMAD(EM1.EV1,X3,X2)
      CALL MATMLT(EM1,XD3,XD2)
C
C----ITH=0 ON FIRST TIME THROUGH LOOP
C    ITH=N ON NTH TIME THROUGH LOOP
C    ITH=-1 ON LAST TIME THROUGH LOOP
      IF(ITH.GE.0)      GO TO 14
C
C----SET UP INDICES AND VECTORS FOR LAST TIME THROUGH LOOP
```

```
  125 DO 13 I=1,3                                                       LENT1930
      WH(I)=WT(I)                                                       LENT1940
      CALL MATMLT(EM1,WH,WR)                                            LENT1950
      X1(I)=X2(I)                                                       LENT1960
   13 XD1(I)=XD2(I)                                                     LENT1970
      JTH=8                                                             LENT1980
      KTH=32                                                            LENT1990
      ITH=-1                                                            LENT2000
      GO TO 15                                                          LENT2010
C                                                                       LENT2020
C----GENERATE A TOTAL OF MWLD "WILD" POINTS BEGINNING AT TIME TWLD     LENT2030
C    NWLD SET EQUAL TO MWLD AND DECREMENTED UNTIL NEGATIVE              LENT2040
C    AT THAT TIME WILD POINTS STOP                                      LENT2050
  14  IF(T.NE.TWLD)     GO TO 141                                       LENT2060
      NWLD=MWLD                                                         LENT2070
  141 NWLD=NWLD-1                                                       LENT2080
C                                                                       LENT2090
      CALL RADAR9                                                       LENT2100
C                                                                       LENT2110
C----COMPUTE RADAR ESTIMATION ERROR RESIDUALS                           LENT2120
      DUMSUM=0.D0                                                       LENT2130
      IF(ITH.EQ.0) WRITE(6,8)                                           LENT2140
      NSAMPL=NSAMPL+1                                                   LENT2150
      DO 142 I=1,3                                                      LENT2160
      FILRES(I)=X1(I)-X2(I)                                             LENT2170
      DUMSUM=DUMSUM+FILRES(I)**2                                        LENT2180
  142 SUMSQR(I)=SUMSQR(I)+FILRES(I)**2                                  LENT2190
      FILRES(4)=DSQRT(DUMSUM)                                           LENT2200
      SUMSQR(4)=SUMSQR(4)+DUMSUM                                        LENT2210
      IF(ITH.EQ.0) GO TO 18                                             LENT2220
C                                                                       LENT2230
C----X7 CONTAINS ESTIMATE OF TARGET-AIRCRAFT COORDINATES                LENT2240
C     IN THE RADAR FRAME.                                               LENT2250
   15 DO 16 I=1,3                                                       LENT2260
   16 X7(I)=EV1(I)-X1(I)                                                LENT2270
C                                                                       LENT2280
C----X5 CONTAINS COORDINATES OF THE TARGET IN THE AIRCRAFT              LENT2290
C     (PRIMED) REFERENCE FRAME.                                         LENT2300
      CALL MATMLT(EM2,X7,X5)                                            LENT2310
      CALL MATMLT(EM2,XD1.XD5)                                          LENT2320
C                                                                       LENT2330
      CA=DSQRT(XD5(1)*XD5(1)+XD5(2)*XD5(2))                             LENT2340
      SA=XD5(1)/CA                                                      LENT2350
      CA=XD5(2)/CA                                                      LENT2360
      EM3(1,1)=CA                                                       LENT2370
      EM3(1,2)=-SA                                                      LENT2380
      EM3(2,1)=SA                                                       LENT2390
      EM3(2,2)=CA                                                       LENT2400
```

```
C                                                                       LENT2410
C----X6 CONTAINS COORDINATES OF THE TARGET IN THE AIRCRAFT FRAME,       LENT2420
C    ROTATED SUCH THAT THE XD6(2) VECTOR POINTS AT THE TARGET.          LENT2430
C    NOTE: XD5 AND XD6 ARE ACTUALLY NEGATIVE DERIVATIVES OF XD5/XD6     LENT2440
      CALL MATMLT(EM3,X5,X6)                                            LENT2450
      CALL MATMLT(EM3,XD5,XD6)                                          LENT2460
C                                                                       LENT2470
      AA=DEG*DARCOS(CA)                                                 LENT2480
      EM3(1,2)=-EM3(1,2)                                                LENT2490
      EM3(2,1)=-EM3(2,1)                                                LENT2500
C                                                                       LENT2510
      X4(1)=0.D0                                                        LENT2520
      X4(2)=XD6(2)                                                      LENT2530
      X4(3)=XD6(3)                                                      LENT2540
C                                                                       LENT2550
C----PUT DATA BACK INTO UNROTATED A/C COORDINATES                       LENT2560
      CALL MATMLT(EM3,X4,XD5)                                           LENT2570
C                                                                       LENT2580
      HATS=HAT                                                          LENT2590
      IF(IB1.EQ.0.OR.IB1.EQ.2)      HAT=-X6(3)                          LENT2600
C                                                                       LENT2610
C----BYPASS DIVE EQUATIONS WHEN IB1=0                                   LENT2620
      IF(IB1.NE.1)        GO TO 165                                     LENT2630
      T2=XD6(2)*CD2                                                     LENT2640
      DY2=XD6(2)*XD6(2)                                                 LENT2650
      DZ2=CD4*DY2                                                       LENT2660
      DY2=CD3*DY2                                                       LENT2670
      V2=XD6(2)+ATH*T2                                                  LENT2680
      V3=V2+ATH*T3                                                      LENT2690
      DY3=.5D0*T3*(V2+V3)                                               LENT2700
      DZ3=DY3*STHD                                                      LENT2710
      DY3=DY3*CTHD                                                      LENT2720
      VH=V3*CTHD                                                        LENT2730
      VV=-V3*STHD                                                       LENT2740
      HAT=-X6(3)-DZ2-DZ3                                                LENT2750
      X4(1)=VH*SA-WH(1)                                                 LENT2760
      X4(2)=VH*CA-WH(2)                                                 LENT2770
      X4(3)=VV                                                          LENT2780
      GO TO 170                                                         LENT2790
C                                                                       LENT2800
C----SUBTRACT EST WIND VALUES AT ALT OF TARGET IN A/C COORDINATES       LENT2810
  165 DO 17 I=1,3                                                       LENT2820
   17 X4(I)=XD5(I)-WH(I)                                                LENT2830
  170 VE2=X4(1)*X4(1)+X4(2)*X4(2)                                       LENT2840
C                                                                       LENT2850
C----VEH IS HORIZONTAL AIRSPEED                                         LENT2860
C    VE IS TOTAL AIRSPEED                                               LENT2870
C    DOWNWARD VELOCITY ANGLE(THE) IS SET TO ZERO IN LEVEL BOMB MODE     LENT2880
```

```
C     HA IS TOTAL HEIGHT OF TARGET ABOVE SEA LEVEL                      LENT2890
      VEH=DSQRT(VE2)                                                    LENT2900
      SG=X4(1)/VEH                                                      LENT2910
      CG=X4(2)/VEH                                                      LENT2920
      SAG=SA*CG-CA*SG                                                   LENT2930
      CAG=CA*CG+SA*SG                                                   LENT2940
      IF(IB1.NE.2.OR.XD6(3).GE.-.100*VEH)      GO TO 171                LENT2950
      IB1=3                                                             LENT2960
      TP=T                                                              LENT2970
      HAT=HATS                                                          LENT2980
  171 VE2=VE2+X4(3)*X4(3)                                               LENT2990
      VE=DSQRT(VE2)                                                     LENT3000
      THE=0.D0                                                          LENT3010
      IF(ITH.LT.0.OR.IB1.GT.1)     THE=DATAN2(X4(3),VEH)*DEG            LENT3020
      IF(IB1.EQ.0.AND.ITH.GE.0)     X4(3)=0.D0                          LENT3030
      HA=HAT+HT                                                         LENT3040
      VDR(1,2)=HA                                                       LENT3050
      VDR(1,3)=VEH                                                      LENT3060
      VDR(1,4)=X4(3)                                                    LENT3070
C                                                                       LENT3080
C----BRANCH AROUND TIME OF FALL(TF) AND BALLISTIC RANGE(RA)             LENT3090
C     EQUATIONS WHEN TOTAL ELAPSED TIME IS LESS THAN 2 SECONDS          LENT3100
      IF(ITH.LT.0) GO TO 173                                            LENT3110
      IF(ITH.LE.48)     GO TO 18                                        LENT3120
C                                                                       LENT3130
C----TF AND RA ARE COMPUTED ON EACH LOOP, THROUGH THE USE OF PARTIAL    LENT3140
C     DERIVATIVES WHICH ARE CALCULATED EVERY 4 SECONDS, STARTING AT T=2 LENT3150
  173 TF=TFTF+(HA-DSV)*SENCO(1)+(VEH-DXSV)*SENCO(2)+(X4(3)-DZSV)*SENCO(3LENT3160
     1)                                                                 LENT3170
      RA=RARA+(HA-DSV)*SENCO(4)+(VEH-DXSV)*SENCO(5)+(X4(3)-DZSV)*SENCO(6LENT3180
     1)                                                                 LENT3190
   18 JTH=JTH+1                                                         LENT3200
      IF(JTH.EQ.9)     JTH=1                                            LENT3210
      KTH=KTH+1                                                         LENT3220
C                                                                       LENT3230
C----BRANCH AROUND INTERGRATOR UNLESS 4 SECONDS HAVE ELAPSED SINCE LAST LENT3240
      IF(KTH.LT.33)     GO TO 205                                       LENT3250
      KTH=1                                                             LENT3260
C                                                                       LENT3270
C----STORE INITIAL CONDITIONS FOR NEXT 4 SECONDS EXTRAPOLATION          LENT3280
      DSV=ZSV                                                           LENT3290
      DXSV=XDSV                                                         LENT3300
      DZSV=ZDSV                                                         LENT3310
      RARA=FSRA                                                         LENT3320
      TFTF=TSTF                                                         LENT3330
C                                                                       LENT3340
C----STORE PARTIAL VALUES FOR NEXT 4 SECONDS EXTRAPOLATION              LENT3350
      DO 183 I=1,6                                                      LENT3360
```

```
  183 SENCO(I)=VDR(4,I)
      ZSV=HA
      XDSV=VDR(1,3)
      ZDSV=VDR(1,4)
      VDR(1,1)=0.D0
      DO 19 I=5,15
   19 VDR(1,I)=0.D0
      VDR(1, 8)=1.D0
      VDR(1,12)=1.D0
      VDR(1,16)=1.D0
      H=DSQRT(HA/10.D0)/ENSTP
      X=0.D0
C
      CALL STIFF(DER,OUT)
C
      RSRA=VDR(1,1)
      TSTF=X
      JTH=JTH-1
      KTH=KTH-1
      IF(ITH.GE.0)      GO TO 173
C
      RA=RSRA
      TF=X
C
C----ON INITAL 6 SECONDS OF RUN, BRANCH BACK TO BEGINNING
  205 IF(ITH.GE.48.OR.ITH.LT.0) GO TO 2050
      GO TO 306
C
C----BALLISTIC WIND CALCULATIONS
 2050 WBX=WBLX*CA-WBLY*SA
      WBY=WBLX*SA+WBLY*CA
      P=0.D0
      OMP=1.D0
C
C----BYPASS DIVE EQUATIONS
      IF(IB1.NE.2.AND.IB1.NE.3)      GO TO 20500
      P=(HAT-HPF)/(HO-HPF)
      IF(P.GT.1.D0)      P=1.D0
      IF(P.LT.0.D0)      P=0.D0
      OMP=1.D0-P
      TF=P*TPF+OMP*TF
      RA=P*RAF+OMP*RA
      IF(IB1.EQ.2)      HAT=HPF
C
C----COMPUTE BOMB IMPACT POINT.  XGC,YGC, AND ZGC
C     ARE THE IMPACT POINT COORDINATES CORRECTED FOR
C     CORRECLIS ACCELERATION
20500 XG=WBX*TF-RA*SAG
```

```
      IF(TF.LT.0.000001)TF=0.000001
      YG=WBY*TF+RA*CAG
      DXG=WE*TF*(HAT*CPH*CA+RA*SPH)
      DYG=WE*TF*HAT*CPH*SA
      DZG=WE*TF*RA*CPH*SA
      XGC=XG+DXG
      YGC=YG+DYG
      ZGC=DZG
      IF(IB1.GT.0)      GO TO 2051
C
C----COMPUTE TIME TO GO TO RELEASE BOMB(TG) AND LATERAL ERROR(XE)
      TG=(X6(2)-YGC)/XD6(2)
      XEO=XE
      XE=X6(1)-XGC
      IF(ITH.EQ.48) XELIM=.05*XE
      IF(XE.LE.XELIM) IIFLG=1
      IF(IIFLG.EQ.1) XEEXPN=XE
      GO TO 2059
C
C----FOLLOWING BLOCK OF EQUATIONS BYPASSED UNLESS IN DIVE MODE
 2051 IF(IB1.GT.1)      GO TO 2052
      XE=X6(1)-XGC
      TP=(X6(2)-YGC-DY2-DY3)/XD6(2)
      TG=TP+T2+T3
      IF(TP.GT.0.D0)      GO TO 2059
      IB1=2
      TP=T
      TPF=TF
      RAF=RA
      HPF=HAT
      HO=-X6(3)
      THC=THD
      GO TO 2059
 2052 IF(IB1.GT.2)      GO TO 2053
      TG=(X6(2)-YGC)/XD6(2)
      GO TO 2054
 2053 TG=(X6(3)+HAT)/XD6(3)
 2054 YPPA=TG*XD6(2)
      ZPPA=TG*XD6(3)
      YPP1=YPPA+YGC
      XPPM=X6(1)-XGC
      YPPM=X6(2)-YPP1
      DTG=YPPM/(XD6(2)+XD6(3)*OMP*(SENCO(4)*CAG+WBY*SENCO(1)))
      TG=TG+DTG
      XE=XPPM-DTG*XD6(3)*OMP*(-SENCO(4)*SAG+WBX*SENCO(1))
      IF(IB1.GT.2)      GO TO 2055
      HAT=-X6(3)+XD6(3)*TG
      GO TO 2056
```

```
 2055 HAT=HAT+XD6(3)*DTG                                                  LENT4330
 2056 IF(IB1.EQ.4)      GO TO 2059                                        LENT4340
      IF((HAT-HPF).GT.HTOL.AND.(TTHD+XD6(3)/XD6(2)).GT.ATTL)    GOTO2059LENT4350
      IB1=4                                                               LENT4360
      TP=T                                                                LENT4370
      THC=DEG*DATAN(-XD6(3)/XD6(2))                                       LENT4380
C                                                                         LENT4390
C----X6 AND XD6 AT THIS POINT ARE ERRORS IN ESTIMATED A/C                 LENT4400
C    POSITION AND VELOCITY IN ROTATED A/C SYSTEM                          LENT4410
 2059 DO 206 I=1,3                                                        LENT4420
      X5(I)=-X5(I)                                                        LENT4430
      X6(I)=X3(I)-X5(I)                                                   LENT4440
  206 XD6(I)=XD3(I)-XD5(I)                                                LENT4450
C                                                                         LENT4460
C----RBT IS HORIZONTAL DISTANCE BETWEEN A/C AND TARGET                    LENT4470
C    HDE IS THE ANGLE BETWEEN BOMB IMPACT AND THE TARGET IN DEGREES       LENT4480
C    HDEDOT IS THE ANGLE ERROR RATE                                       LENT4490
C    CTV IS CROSS TRACK VELOCITY                                          LENT4500
      RBT=DSQRT(X5(1)*X5(1)+X5(2)*X5(2))                                  LENT4510
      HDEOLD=HDE                                                          LENT4520
      HDE=DARSIN(XE/RBT)*DEG                                              LENT4530
      HDEDOT=(HDE-HDEOLD)/DT                                              LENT4540
      IF(DABS(T-6.).LE.0.0001) HDEDOT=0.D0                                LENT4550
C                                                                         LENT4560
C----IF T IS LESS THAN 20 SECS AND TG IS LESS THAN 2 SECS,                LENT4570
C    SET A DUMMY TIME TO GO(TQ) EQUAL TO 3 SECS                           LENT4580
      TQ=TG                                                               LENT4590
      IF(T.GT.20.0) GO TO 207                                             LENT4600
      IF(TQ.LE.2.) TQ=3.                                                  LENT4610
  207 IF(ITH.LT.0)      GO TO 209                                         LENT4620
C                                                                         LENT4630
C----IF TG IS LESS THAN ZERO, BRANCH FOR LAST TIME THROUGH LOOP           LENT4640
      IF(TG.LE.0.D0)      GO TO 125                                       LENT4650
      IF(IB1.GT.0.AND.IB1.LT.4)      TG=1000.D0                           LENT4660
      CTV=-XE*VEH/RBT                                                     LENT4670
      DXE=(XE-XEO)/DT                                                     LENT4680
      IF(DABS(T-6.).LE.0.0001) DXE=0.D0                                   LENT4690
C                                                                         LENT4700
C----GENERATE CONTROL COMMAND BANK ANGLE                                  LENT4710
C    PHD1 IS COMPONENT BASED ON ANGLE ERROR HDE                           LENT4720
C    PHD2 IS COMPONENT BASED ON ANGLE ERROR RATE HDEDOT                   LENT4730
C    PHD IS REQUESTED COMMAND ANGLE IN DEGREES                            LENT4740
C    GENERATE NEW COMMAND EVERY DTCON SECONDS                             LENT4750
      IF(MOD(ITH,IDTCON).NE.0) GO TO 2082                                 LENT4760
      PHD1=G1*HDE                                                         LENT4770
      PHD2=G2*HDEDOT                                                      LENT4780
      PHD=PHD1+PHD2                                                       LENT4790
C                                                                         LENT4800
```

```
C----ILVL IS INITIALIZED TO ZERO AND SET TO 1 ONLY AFTER TIME TG
C     IS FOUND LESS THAN TLVL, WHICH IS THE TIME AT THE END ON MISSION
C     FOR WHICH NO COMMANDS ARE TO BE SENT
2082  IF(ILVL.EQ.1)        GO TO 2087
      IF(TQ.LE.TLVL)        GO TO 20855
      IF(TQ.GT.TLVL1)        GO TO 2086
      PHDAVG=PHDAVG+PH1
      JAVG=JAVG+1
      GO TO 2086
20855 ILVL=1
      PHD=PHDAVG/FLOAT(JAVG)
C
C----COMPUTE BANK COMMAND, PHC TO NEAREST 15/128 OF A DEGREE
C     PHC IS LIMITED TO PLUS OR MINUS 30 DEGREES
 2086 PHC=(PHD)*128.D0/15.D0+.5D0
      IF(PHC.LT.0.D0)       PHC=PHC-1.D0
      NHC=PHC
      PHC=NHC
      PHC=PHC*15.D0/128.D0
      IF(PHC.GT.29.8828125)       PHC=29.8828125
      IF(PHC.LT.-30.D0)       PHC=-30.D0
C
C----IF COMPUTED TIME TO GO IS LESS THAN THE SAMPLING INTERVAL
C     SET TG=TQ
2087  IF(TQ.LT.DT)       ITH=-2
      IF(TQ.LT.DT)       TG=TQ
C
C----GO TO 209 TO PRINT OUTPUT FOR THIS TIME THROUGH LOOP
      IF(NWLD.GT.-2)       GO TO 209
 306  IMOD=IMOD+1
      IF(TG.LE.1.0) IMOD=8
      IF(IMOD.NE.8) GO TO 12
      IMOD=0
      IF(ITH.LT.48) GO TO 209
C
C----STORE VALUES FOR PLOTTING EVERY SECOND
      ITAB1=ITAB1+1
      XXA(ITAB1)=T
      XXB(ITAB1)=FILRES(1)
      XXC(ITAB1)=FILRES(2)
      XXD(ITAB1)=FILRES(3)
      XXE(ITAB1)=FILRES(4)
      XXF(ITAB1)=X6(1)
      XXG(ITAB1)=X6(2)
      YYA(ITAB1)=XE
      YYB(ITAB1)=XEEXPN
      YYC(ITAB1)=DXE
      YYD(ITAB1)=PHD
```

```
      YYE(ITAB1)=PHD1
      YYF(ITAB1)=PHD2
      YYG(ITAB1)=PHC
      YYH(ITAB1)=PH1
      YYO(ITAB1)=PH
C     YYP(ITAB1)=
      YYQ(ITAB1)=XDD1(1)
      YYR(ITAB1)=XDD1(2)
C     YYS(ITAB1)=
C     YYS1(ITAB1)=
C     YYS2(ITAB1)=
C     YYT(ITAB1)=
C     YYU(ITAB1)=
C
C----PRINT SUMMARY OF VALUES EACH SECOND
 209  PRINT 4, X1,XD1,XDD1(1),XDD1(2),
     1          X2,XD2,TF,RA,
     2          X3,XD3,XGC,YGC,
     3          X5,XD5,AA,HAT,
     4          X6,XD6,YNP,ZIP,
     5          PHD1,PHD2,PHD,HDE,HDEDOT,TG,VEH,THN,
     6          PH,PH1,PHC,PS,PS1,ZIP,ZIP,ZIP,
     7          XE,DXE,CTV,ZIP,GNX,GNY,GNZ,T
C----LNC KEEPS TRACK OF NUMBER OF OUTPUTS ON A PAGE, AND THE WRITE
C    STATEMENT YIELDS A NEW PAGE
      IF(ITH.EQ.-1) GO TO 22
      LNC=LNC+1
      IF(LNC.LT.7) GO TO 21
      LNC=0
      WRITE(6,8)
C----LIMIT OF 2000 ITERATIONS PER PROBLEM
   21 IF(ITH.LE.2000)      GO TO 12
C
C----CALCULATE AND PRINT FINAL ERROR SUMMARY, RESIDUAL VALUES
C    AND BIAS ESTIMATE AT PROBLEM TERMINATION
  22  X1(1)=XGC
      X1(2)=YGC
      X1(3)=0.D0
      X3(3)=0.D0
      CALL MATMAD(EM3,X3,X1,X2)
      X2(3)=DSQRT(X2(1)*X2(1)+X2(2)*X2(2))
      DO 1010 I=1,4
1010  SUMSQR(I)=DSQRT(SUMSQR(I)/NSAMPL)
      PRINT 7,X2,SUMSQR
C
C----EXECUTE PLOTS
      WRITE(6,991) ATITLE
      CALL PLOTP(XXA,XXB,ITAB1,0)
```

```
      WRITE(6,991) BTITLE                                               LENT5770
      CALL PLOTP(XXA,XXC,ITAB1,0)                                       LENT5780
      WRITE(6,991) CTITLE                                               LENT5790
      CALL PLOTP(XXA,XXD,ITAB1,0)                                       LENT5800
      WRITE(6,991) DTITLE                                               LENT5810
      CALL PLOTP(XXA.XXE,ITAB1,0)                                       LENT5820
      WRITE(6,991) ETITLE                                               LENT5830
      CALL PLOTP(XXA,XXF,ITAB1:0)                                       LENT5840
      WRITE(6,991) FTITLE                                               LENT5850
      CALL PLOTP(XXA,XXG,ITAB1,0)                                       LENT5860
      WRITE(6,991) GTITLE                                               LENT5870
      CALL PLOTP(XXF,XXG,ITAB1,0)                                       LENT5880
      WRITE(6,991) HTITLE                                               LENT5890
      CALL PLOTP(XXA,YYA,ITAB1,0)                                       LENT5900
      WRITE(6,991) OTITLE                                               LENT5910
      CALL PLOTP(XXA,YYB,ITAB1,0)                                       LENT5920
      WRITE(6.991) PTITLE                                               LENT5930
      CALL PLOTP(XXA,YYC,ITAB1,0)                                       LENT5940
      WRITE(6,991) QTITLE                                               LENT5950
      CALL PLOTP(XXA,YYD,ITAB1,1)                                       LENT5960
      CALL PLOTP(XXA,YYE,ITAB1,2)                                       LENT5970
      CALL PLOTP(XXA,YYF,ITAB1,3)                                       LENT5980
      WRITE(6,991) RTITLE                                               LENT5990
      CALL PLOTP(XXA,YYG,ITAB1,1)                                       LENT6000
      CALL PLOTP(XXA,YYH,ITAB1,2)                                       LENT6010
      CALL PLOTP(XXA,YYO,ITAB1,3)                                       LENT6020
      WRITE(6,991) TTITLE                                               LENT6030
      CALL PLOTP(XXA,YYQ,ITAB1,0)                                       LENT6040
      WRITE(6,991) UTITLE                                               LENT6050
      CALL PLOTP(XXA,YYR,ITAB1,0)                                       LENT6060
      WRITE(6,991) VTITLE                                               LENT6070
      CALL PLOTP(YYQ,YYR,ITAB1,0)                                       LENT6080
C     WRITE(6,991) WTITLE                                               LENT6090
C     WRITE(6,991) XTITLE                                               LENT6100
C     WRITE(6,991) YTITLE                                               LENT6110
C     WRITE(6,991) ZTITLE                                               LENT6120
      PRINT 992                                                         LENT6130
  999 CONTINUE                                                          LENT6140
      STOP                                                              LENT6150
C                                                                       LENT6160
C----INPUT FORMAT STATEMENTS                                            LENT6170
    1 FORMAT(12F6.0)                                                    LENT6180
C                                                                       LENT6190
    2 FORMAT(6F10.3)                                                    LENT6200
C                                                                       LENT6210
    3 FORMAT(6F10.3,I3)                                                 LENT6220
C                                                                       LENT6230
  101 FORMAT(6A8)                                                       LENT6240
```

```
C
 102  FORMAT(5F10.3,2I3)
C
C----OUTPUT FORMAT STATEMENTS
C
    4 FORMAT(/2X,  8D16.5/(2X,8D16.5))
C
    5  FORMAT(1H1,48X,'AN/TPQ-27 SIMULATION',//,37X,'PRECISION ',
     1'GUIDANCE MODE WITH KALMAN FILTERING',///,5X,'INITIAL CONDITIONS :
     2',//,8X,'TRUE WIND AT TARGET = ',10X,1P3D16.5,/,8X,
     3'ESTIMATED WIND AT TARGET = ',5X,1P3D16.5,//,8X,
     4'RADAR DATA',/,10X,
     5'NUMBER OF NOISY POINTS(NWLD) = ',I5,/,10X,
     C'START TIME OF WILD POINTS = ',5X,0PF7.3,/10X,
     6'MEASUREMENT SIGMAS(R(FT.),AZ(MRAD),EL(MRAD)) = ',1P3D16.5,/,10X
     7,'MEASUREMENT BIASES(            "              ) = ' ,1P3D16.5,/,
     810X,'INITIAL VELOCITY MEASUREMENT VALUES = ',9X,1P3D16.5,/,
     910X,'RANDOM FORCING ASSUMPTION VALUES (SIGW) = ',5X,1P3D16.5,/,
     110X,'RADAR SAMPLING INTERVAL = ',26X,0PF6.4,//,8X,
     2'INITIAL POSITION OF A/C IN TARGET SYSTEM = ',1P3D16.5,/,8X,
     3'INITIAL VELOCITY OF A/C IN TARGET SYSTEM = ',1P3D16.5,///)
C
   60  FORMAT(5X,'AIRCRAFT PARAMETERS :',//,9X,'TB = ',F8.5,6X,'PHB = ',
     1F8.5,///)
C
   61  FORMAT(5X,'CONTROL PARAMETERS:',//,9X,'DTCON = ',
     1F8.5,/,9X,'G1 = ',F8.4,/,9X,'G2 = ',F8.4,///)
C
   62  FORMAT(5X,'BALLISTIC TABLE PARAMETERS:',//,9X,
     1'BALLISTIC WIND VALUES (WBLX,WBLY) = ',2F10.6,/,9X,
     2'BFF = ',F10.6,/,9X,'D = ',2X,F10.6,/,9X,
     3'W = ',2X,F10.6,/,9X,'NTB = ',I5,///)
C
   63  FORMAT(5X,'DIVE BOMBING MODE PARAMETERS:',//,9X,
     1'THD = ',F8.2,6X,'ATT = ',F8.2,/,9X,
     2'ATH = ',F8.2,6X,'ANH = ',F8.2,/,9X,
     3'TUP = ',F8.2,5X,'HTOL = ',F8.2,/,8X,
     4'ATOL = ',F8.2,5X,'THNM = ',F8.2,/,8X,
     5'THDM = ',F8.2,6X,'IB1 = ',I5,/,8X,
     6'IACC = ',I5,///)
C
    8  FORMAT(1H1,//,59X,'OUTPUT FORMAT',//,
     111X,'X1',14X,'Y1',14X,'Z1',13X,'XD1',13X,'YD1',13X,'ZD1',12X,
     2'XDD1',12X,'YDD1',/,
     311X,'X2',14X,'Y2',14X,'Z2',13X,'XD2',13X,'YD2',13X,'ZD2',12X,
     4'  TF',12X,'  RA',/,
     511X,'X3',14X,'Y3',14X,'Z3',13X,'XD3',13X,'YD3',13X,'ZD3',13X,
     6'XGC',13X,'YGC',/,
```

```
     711X,'X5',14X,'Y5',14X,'Z5',13X,'XD5',13X,'YD5',13X,'ZD5',14X,
     8'AA',13X,'HAT',/,
     911X,'X6',14X,'Y6',14X,'Z6',13X,'XD6',13X,'YD6',13X,'ZD6',10X,
     C3X,'YNP',/,
     19X,'PHD1',12X,'PHD2',13X,'PHD',13X,'HDE',10X,'HDEDOT',14X,'TG',
     213X,'VEH',13X,'THN',/,
     311X,'PH',13X,'PH1',13X,'PHC',14X,'PS',13X,'PS1',/,
     411X,'XE',13X,'DXE',13X,'CTV',10X,'          ',10X,
     5'G(1,1)',10X,'G(4,2)',10X,'G(7,3)',15X,'T',//)
C
    9 FORMAT(26I3)
C
    7  FORMAT(//25X,4HXI  =1PD13.5,6X,4HYI  =D13.5,6X,4HRI  =D13.5,
     1///,20X,'AVG X RESIDUAL = ',0PF10.4,5X,'AVG Y RESIDUAL = ',
     2F10.4,5X,'AVG Z RESIDUAL = ',F10.4,//,49X,
     3'AVG RADIAL RESIDUAL = ',F10.4, ///)
C
   90 FORMAT(//49X,I6,27H IS AN ILLEGAL TABLE NUMBER)
C
  991  FORMAT(1H1,10X,6A8,//,11X,6A8,///)
C
  992  FORMAT(1H1)
C
       END
C
C
C
C
C
C
C
C
       SUBROUTINE ARCRFT
C
       IMPLICIT REAL*8 (A-H,O-Z)
C
       COMMON/AIRCOM/X3(3),XD3(3),WT(3),PHB,TB,DT,PHC,
     1               TG,ITH,IB1,PH,PSD,PS,THC,ANH,ATT,TUP,THN,T
C
       DATA DEG,G/57.2957795130823 21,32.174049/
C
C----FIRST TIME THROUGH, ITH=0
       IF(ITH)5,1,4
C
C----SUBTRACT WIND VELOCITY
    1  SM1=XD3(1)-WT(1)
       SM2=XD3(2)-WT(2)
C
```

```
C----VT IS AIRSPEED OF A/C
C     PHB IS AUTOPILOT BANK ANGLE BIAS
C     TB IS A/C RESPONSE LOOP TIME CONSTANT
C     PS IS INITIAL TURN ANGLE
C     PSD IS INITIAL TURN RATE
      VT=DSQRT(SM1*SM1+SM2*SM2)
      PH=PHB
      CA1=G/VT
      CA2=DEXP(-DT/TB)
      CA3=CA1
      CA4=DT*DT/(12.D0*DEG)
      CA5=TB*(1.D0-CA2)
      DT3=.5D0*DT
      PSD=CA1*PH
      PS=DATAN2(SM1,SM2)
      CPS=DCOS(PS)
      SPS=DSIN(PS)
      PS=PS*DEG
C
C----EQUATIONS AND CONSTANTS FOR DIVE MODE FOLLOW
      AT1=ATT*DT3
      VTC=VT
      VTH=VT
      TH=0.D0
      THD=TH
      THN=TH
      THDN=TH
      STH=TH
      STHN=TH
      CTH=1.D0
      CTHN=CTH
      OTP=1.D0/TUP
      CA6=DEXP(-DT/TUP)
      AND=ANH*DEG
      AN8=AND*DT
      RETURN
C
C************************************************************************
C----ENTER MAIN CALCULATION STREAM
C
C----PHC IS NOW THE COMMAND BANK ANGLE AT T(N-1)
C     PHN IS THE NEW BANK ANGLE
C     PSDN IS THE NEW TURNING RATE
C     PSN IS NEW HEADING ANGLE FOR VELOCITY WITH RESPECT TO WIND
    4 T=T+DT
      PSN=PHB+PHC
      PHN=PSN+(PH-PSN)*CA2
      PSDN=CA1*PHN
```

```
      PSN=PS+CA3*(PSN*DT+CA5*(PH-PSN))
      CPSN=PSN/DEG
      SPSN=DSIN(CPSN)
      CPSN=DCOS(CPSN)
C
C----BYPASS FOLLOWING BLOCK OF EQUATIONS UNLESS IN DIVE MODE
      IF(IB1.LE.1)      GO TO 43
      VTO=VT
      VTH=VT+AT1
      VT=VTH+AT1
      CA1=G/VT
      CA3=G/VTH
      THDN=AND/VT
      THN=(THC-TH)*OTP
      IF(DABS(THN).GT.THDN)      GO TO 41
      THDN=THN*CA6
      THN=THC-TUP*THDN
      GO TO 42
   41 THN=AN8/VTH+TH
   42 CTHN=THN/DEG
      STHN=DSIN(CTHN)
      CTHN=DCOS(CTHN)
C
C----UPDATE VELOCITY VECTOR FOR NEW TURN ANGLE
   43 XD3(1)=WT(1)+VT*SPSN*CTHN
      XD3(2)=WT(2)+VT*CPSN*CTHN
      XD3(3)=-VT*STHN
C
C----UPDATE POSITION VECTOR FOR NEW TURN ANGLE
      SM1=DT3*(SPS*CTH+SPSN*CTHN)-CA4*(CPSN*CTHN*PSDN-SPSN*STHN*THDN-CPS
     1    *CTH*PSD+SPS*STH*THD)
      SM2=DT3*(CPS*CTH+CPSN*CTHN)-CA4*(-SPSN*CTHN*PSDN-CPSN*STHN*THDN+
     1    SPS*CTH*PSD-CPS*STH*THD)
      SM3=DT3*(STH+STHN)-CA4*(CTHN*THDN-CTH*THD)
C
      X3(1)=DT*WT(1)+VTH*SM1+X3(1)
      X3(2)=DT*WT(2)+VTH*SM2+X3(2)
      X3(3)=-VTH*SM3+X3(3)
C
C----RESET VALUES FOR NEXT TIME IN SUBROUTINE
      PH=PHN
      PS=PSN
      PSD=PSDN
      SPS=SPSN
      CPS=CPSN
C
      TH=THN
      THD=THDN
```

```
      STH=STHN                                                          LENT8170
      CTH=CTHN                                                          LENT8180
      RETURN                                                            LENT8190
C                                                                       LENT8200
C----ENTER THIS SECTION ONLY ON LAST TIME THROUGH MAIN LOOP             LENT8210
C     RECOMPUTE CONSTANTS FOR LAST DELTAT                               LENT8220
    5 DT=TG                                                             LENT8230
      DT3=.5D0*DT                                                       LENT8240
      AN8=AND*DT                                                        LENT8250
      CA6=DEXP(-DT/TUP)                                                 LENT8260
      CA2=DEXP(-DT/TB)                                                  LENT8270
      CA5=TB*(1.D0-CA2)                                                 LENT8280
      CA4=DT*DT/(12.D0*DEG)                                             LENT8290
      AT1=ATT*DT2                                                       LENT8300
      GO TO 4                                                           LENT8310
      END                                                               LENT8320
C                                                                       LENT8330
C                                                                       LENT8340
C                                                                       LENT3350
C                                                                       LENT8360
C                                                                       LENT8370
C                                                                       LENT8380
C                                                                       LENT8390
C                                                                       LENT8400
      SUBROUTINE RADAR9                                                 LENT8410
C                                                                       LENT8420
C----THIS SUBROUTINE SIMULATES A RADAR PROCESSOR.  IT CONVERTS FROM     LENT8430
C    CARTESIAN TO SPHERICAL, ADDS NOISE, AND THEN CONVERTS BACK TO      LENT8440
C    CARTESIAN COORDINATES.  DATA IS FILTERED USING A                   LENT8450
C    THIRD ORDER KALMAN FILTER WITH DETERMINSITIC FORCING FROM THE      LENT8460
C    AIRCRAFT CONTROLLER INCLUDED.                                      LENT8470
C                                                                       LENT8480
      IMPLICIT REAL*8 (A-H,O-Z)                                         LENT8490
      REAL*4 RAN(3)                                                     LENT8500
C                                                                       LENT8510
      COMMON/AIRCOM/X3(3),XD3(3),WT(3),PHB,TB,DT,PHC,                   LENT8520
     1              TG,ITH,IB1,PH,PSD,PS,THC,ANH,ATT,TUP,THN,T          LENT8530
C                                                                       LENT8540
      COMMON/RADCOM/SIG(6),BIS(3),X2(3),XD2(3),X1(3),XD1(3),            LENT8550
     1              SIGW(3),WR(3),PS1,PH1,XDD1(3),GNX,GNY,              LENT8560
     2              GNZ,IU,NWLD                                         LENT8570
C                                                                       LENT8580
      DIMENSION Q(9,9),GAMMA(9,9),ADUM(9,9),BDUM(9,9),W(9,9),           LENT8590
     1PP(9,9),PE(9,9),H(9,9),HT(9,9),R(9,9),PHI(9,9),SIG1(3),           LENT8600
     2PHITRN(9,9),G(9,9),XIDENT(9,9),X1P(3),XD1P(3),XDD1P(3),           LENT8610
     3XDATA(3),DELX(3)                                                  LENT8620
C                                                                       LENT3630
      EQUIVALENCE (XDATA(1),RR),(XDATA(2),A),(XDATA(3),E)               LENT8640
```

```
C
      DATA DEG,GG/57.29577951308232l;32.174049/
      DATA PSDN,PSD1,PHN,PSN,SN1,SN2,SPSN,CPSN,VT1/9*0.DO/
      DATA ANGMIN,ANGMAX/0.0000001,1.5707963/
C
      IF(ITH.NE.0) GO TO 50
C
C----INITIALIZE BANK ANGLE AND CONSTANTS FOR DETERMINISTIC
C     CONTROL MOTION CALCULATIONS
C
      CAA2=DEXP(-DT/TB)
      CA4=DT*DT/(12.D0*DEG)
      CA5=TB*(1.D0-CAA2)
      DT3=DT/2.D0
C
C----GENERATE Q ARRAY
      DO 1 I=1,9
      DO 1 J=1,9
      PE(I,J)=0.D0
      BDUM(I,J)=0.D0
      ADUM(I,J)=0.D0
      PP(I,J)=0.D0
      G(I,J)=0.D0
      R(I,J)=0.D0
      GAMMA(I,J)=0.D0
      W(I,J)=0.D0
    1 Q(I,J)=0.D0
      DT2=DT*DT/2.D0
      DT4=DT**3/6.D0
      W(1,1)=SIGW(1)**2
      W(2,2)=SIGW(2)**2
      W(3,3)=SIGW(3)**2
      GAMMA(1,1)=DT4
      GAMMA(2,1)=DT2
      GAMMA(3,1)=DT
      GAMMA(4,2)=DT4
      GAMMA(5,2)=DT2
      GAMMA(6,2)=DT
      GAMMA(7,3)=DT4
      GAMMA(8,3)=DT2
      GAMMA(9,3)=DT
      CALL TRANS(GAMMA,9,3,ADUM)
      CALL PROD(GAMMA,W,9,3,3,BDUM)
      CALL PROD(BDUM,ADUM,9,3,9,Q)
C
C----INITIALIZE COVARIANCE OF PREDICTION ARRAY (PP)
      DO 2 I=1,9
      DO 2 J=1,9
```

```
      PP(I,J)=0.D0
    2 IF(I.EQ.J) PP(I,J)=1.D06
C
C-----COMPUTE MEASUREMENT MATRIX AND VARIANCE
      DO 3 I=1,9
      DO 3 J=1,9
    3 H(I,J)=0.D0
      H(1,1)=1.D0
      H(2,4)=1.D0
      H(3,7)=1.D0
      CALL TRANS(H,3,9,HT)
C
      SIG1(1)=SIG(1)
      SIG1(2)=SIG(2)*1.D-03
      SIG1(3)=SIG(3)*1.D-03
      VARR=SIG1(1)**2
      VARP=SIG1(2)**2
      VART=SIG1(3)**2
C
C-----COMPUTE STATE TRANSITION ARRAY (PHI)
      DO 4 I=1,9
      DO 4 J=1,9
      PHI(I,J)=0.D0
    4 IF(I.EQ.J) PHI(I,J)=1.D0
      PHI(1,2)=DT
      PHI(1,3)=DT2
      PHI(2,3)=DT
      PHI(4,5)=DT
      PHI(4,6)=DT2
      PHI(5,6)=DT
      PHI(7,8)=DT
      PHI(7,9)=DT2
      PHI(8,9)=DT
      CALL TRANS(PHI,9,9,PHITRN)
C
C-----CREATE IDENTITY ARRAY
      DO 8 I=1,9
      DO 8 J=1,9
      XIDENT(I,J)=0.D0
    8 IF(I.EQ.J) XIDENT(I,J)=1.D0
C
      PRINT 310,((H(I,J),J=1,9),I=1,9)
  310 FORMAT(///,10X,'H ARRAY',//,9(/,9F14.5))
      PRINT 311,((GAMMA(I,J),J=1,9),I=1,9)
  311 FORMAT(1H1,///,10X,'GAMMA ARRAY',//,9(/,9F14.5))
      PRINT 312,((Q(I,J),J=1,9),I=1,9)
  312 FORMAT(///,10X,'Q ARRAY',//,9(/,9F14.5))
      PRINT 313,((PP(I,J),J=1,9),I=1,9)
```

```
  313 FORMAT(///,10X,'PP ARRAY',//,9(/,9F14.5))                     LENT9610
      PRINT 316,((PHI(I,J),J=1,9),I=1,9)                              LENT9620
  316 FORMAT(///,10X,'PHI ARRAY',//,9(/,9F14.4))                      LENT9630
C*********************************************************************LENT9640
C                                                                     LENT9650
C----BEGIN NORMAL FILTER COMPUTATIONS                                 LENT9660
C                                                                     LENT9670
C----COMPUTE TRUE R,AZ,EL,ADD NOISE, AND COMPUTE NOISY MEASUREMENTS   LENT9680
C    IN CARTESIAN COORDINATES                                         LENT9690
   50 IF(NWLD.GE.0.AND.ITH.NE.0) GO TO 61                             LENT9700
      RR=X2(1)*X2(1)+X2(2)*X2(2)                                      LENT9710
      A=DATAN2(X2(1),X2(2))                                           LENT9720
      E=DATAN2(X2(3),DSQRT(RR))                                       LENT9730
      RR=DSQRT(RR+X2(3)*X2(3))                                        LENT9740
      RM2=RR**2                                                       LENT9750
C                                                                     LENT9760
      CALL NORMAL(IU,RAN,3)                                           LENT9770
      DO 6 I=1,3                                                      LENT9780
    6 XDATA(I)=XDATA(I)+BIS(I)+SIG1(I)*RAN(I)                         LENT9790
C                                                                     LENT9800
      IF(DABS(A).LT.ANGMIN) A=ANGMIN                                  LENT9810
      IF(DABS(E).LT.ANGMIN) E=ANGMIN                                  LENT9820
      SIGNA=A/DABS(A)                                                 LENT9830
      SIGNE=E/DABS(E)                                                 LENT9840
      DELA=DABS(A-ANGMAX)                                             LENT9850
      DELE=DABS(E-ANGMAX)                                             LENT9860
      IF(DELA.LT.ANGMIN) A=SIGNA*ANGMAX                               LENT9870
      IF(DELE.LT.ANGMIN) E=SIGNE*ANGMAX                               LENT9880
C                                                                     LENT9890
      CA=DCOS(A)                                                      LENT9900
      SA=DSIN(A)                                                      LENT9910
      CE=DCOS(E)                                                      LENT9920
      SE=DSIN(E)                                                      LENT9930
      CA2=CA**2                                                       LENT9940
      SA2=SA**2                                                       LENT9950
      CE2=CE**2                                                       LENT9960
      SE2=SE**2                                                       LENT9970
      XDATA(3)=RR*SE                                                  LENT9980
      XDATA(2)=RR*CE*CA                                               LENT9990
      XDATA(1)=RR*CE*SA                                               LENT0010
C                                                                     LENT0020
      IF(ITH.NE.0) GO TO 63                                           LENT0030
      DO 62 I=1,3                                                     LENT0040
      X1P(I)=XDATA(I)                                                 LENT0050
      X1(I)=X1P(I)                                                    LENT0060
      XD1P(I)=SIG(I+3)                                                LENT0070
      XDD1P(I)=0.D0                                                   LENT0080
   62 DELX(I)=0.D0                                                    LENT0090
```

```
      IF(NWLD.GE.0) GO TC 61
C
C----COMPUTE COVARIANCE OF MEASUREMENT ERROR ARRAY (R)
   63 R(1,1)=RM2*(VART*SE2*SA2+VARP*CE2*CA2+VART*VARP*SE2*CA2)
     1     +VARR*CE2*SA2
      R(2,2)=RM2*(VART*SE2*CA2+VARP*CE2*CA2+VART*VARP*SE2*SA2)
     1     +VARR*CE2*CA2
      R(3,3)=RM2*VART*CE2+VARR*SE2
      R(1,2)=RM2*VART*(1.DO-VARP)*(SE2*SA*CA)
     1       +(VARR-RM2*VARP)*(CE2*SA*CA)
      R(2,1)=R(1,2)
      R(1,3)=(VARR-RM2*VART)*SE*CE*SA
      R(3,1)=R(1,3)
      R(2,3)=(VARR-RM2*VART)*SE*CE*CA
      R(3,2)=R(2,3)
C
  121 IF(ITH.EQ.0)  PRINT 317,((R(I,J),J=1,9),I=1,9)
  317 FORMAT(///,10X,'R ARRAY ',//,9(/,9F14.5),///)
C
C----COMPUTE GAIN MATRIX
      DO 81 I=1,9
      DO 81 J=1,9
      BDUM(I,J)=0.DO
   81 ADUM(I,J)=0.DO
      CALL PROD(H,PP,3,9,9,ADUM)
      CALL PROD(ADUM,HT,3,9,3,BDUM)
      DO 83 I=1,9
      DO 83 J=1,9
   83 ADUM(I,J)=0.DO
      CALL ADDSUB(BDUM,R,3,3,ADUM,1)
      DO 82 I=1,9
      DO 82 J=1,9
   82 BDUM(I,J)=0.DO
      CALL INVERT(3,ADUM,BDUM,KER,9)
      IF(KER.EQ.2)  PRINT 308
  308 FORMAT(10X,25(1H*),'INVERSION SINGULARITY ENCOUNTERED')
      DO 84 I=1,9
      DO 84 J=1,9
   84 ADUM(I,J)=0.DO
      CALL PROD(PP,HT,9,9,3,ADUM)
      CALL PROD(ADUM,BDUM,9,3,3,G)
      GNX=G(1,1)
      GNY=G(4,2)
      GNZ=G(7,3)
C
C----COMPUTE COVARIANCE OF ESTIMATION ARRAY (PE)
      DO 85 I=1,9
      DO 85 J=1,9
```

```
      ADUM(I,J)=0.DO
   85 BDUM(I,J)=0.DO
      CALL PRGD(G,H,9,3,9,ADUM)
      CALL ADDSUB(XIDENT,ADUM,9,9,BDUM,-1)
      CALL PRGD(BDUM,PP,9,9,9,PE)
C
C----COMPUTE COVARIANCE OF PREDICTION ARRAY (PP)
      DO 86 I=1,9
      DC 86 J=1,9
      ADUM(I,J)=0.DO
   86 BDUM(I,J)=0.DO
      CALL PRCD(PHI,PE,9,9,9,ADUM)
      CALL PRCD(ADUM,PHITRN,9,9,9,BDUM)
      CALL ADDSUB(BDUM,Q,9,9,PP,1)
   61 IF(ITH.LE.1) GO TO 66
C
C----COMPUTE NEW HEADING ANGLE AND HEADING ANGLE RATE, BASED ON
C    UNBIASED COMMANDS AS SENT FROM THE CONTROLLER.
C    PHN IS THE NEW ROLL ANGLE
C    PSDN IS THE NEW TURNING ANGLE RATE
C    PSN IS THE NEW HEADING ANGLE
      PHN=PHC+(PH1-PHC)*CAA2
      CA1=GG/VT1
      PSDN=CA1*PHN
      DELPS1= CA1*(PHC*DT+CA5*(PH1-PHC))
      PS1DEX=DELPS1/DT
      PSN=PS1+DELPS1
C
      CPSN=PSN/DEG
      SPSN=DSIN(CPSN)
      CPSN=DCOS(CPSN)
C
C----COMPUTE STATE PREDICTION VECTOR
   66 XD1P(1)=VT1*SPSN+DT*XDD1(1)
      XD1P(2)=VT1*CPSN+DT*XDD1(2)
      XD1P(3)=XD1(3)+DT*XDD1(3)
C
      DELX(1)=DT3*(XD1P(1)+SN1)-CA4*(XD1P(2)*PSDN-SN2*PSD1)
      DELX(2)=DT3*(XD1P(2)+SN2)-CA4*(-XD1P(1)*PSDN+SN1*PSD1)
      DELX(3)=XD1(3)*DT+XDD1(3)*DT2
C
      DO 110 I=1,3
      XDD1P(I)=XDD1(I)
      XD1P(I)=XD1P(I)+WR(I)
  110 X1P(I)=X1(I)+DT*WR(I)+DELX(I)
C
      IF(NWLD.LT.0) GO  TO 67
      DC 64 I=1,3
```

```
      X1(I)=X1P(I)
      XD1(I)=XD1P(I)
   64 XDD1(I)=XDD1P(I)
      GO TO 65
C
C----COMPUTE STATE ESTIMATION VECTOR
   67 E1=XDATA(1)-X1P(1)
      E2=XDATA(2)-X1P(2)
      E3=XDATA(3)-X1P(3)
C
      X1(1)=X1P(1)+G(1,1)*E1+G(1,2)*E2+G(1,3)*E3
      XD1(1)=XD1P(1)+G(2,1)*E1+G(2,2)*E2+G(2,3)*E3
      XDD1(1)=XDD1P(1)+G(3,1)*E1+G(3,2)*E2+G(3,3)*E3
      X1(2)=X1P(2)+G(4,1)*E1+G(4,2)*E2+G(4,3)*E3
      XD1(2)=XD1P(2)+G(5,1)*E1+G(5,2)*E2+G(5,3)*E3
      XDD1(2)=XDD1P(2)+G(6,1)*E1+G(6,2)*E2+G(6,3)*E3
      X1(3)=X1P(3)+G(7,1)*E1+G(7,2)*E2+G(7,3)*E3
      XD1(3)=XD1P(3)+G(8,1)*E1+G(8,2)*E2+G(8,3)*E3
      XDD1(3)=XDD1P(3)+G(9,1)*E1+G(9,2)*E2+G(9,3)*E3
C
      IF(ITH.GT.1) GO TO 65
      DO 98 I=1,3
   98 XDD1(I)=0.D0
C
C----COMPUTE VALUES FOR ENTERING DETERMINISTIC CONTROL
C    CALCULATIONS ON NEXT ITERATION
   65 IF(ITH.EQ.0) RETURN
      PH1=PHN
      PSD1=PSDN
      SN1=XD1(1)-WR(1)
      SN2=XD1(2)-WR(2)
      VT1=DSQRT(SN1*SN1+SN2*SN2)
      PS1=DATAN2(SN1,SN2)
      CPS=DCOS(PS1)
      SPS=DSIN(PS1)
      PS1=PS1*DEG
C
      RETURN
      END
C
C
C
C
C
C
C
C
      SUBROUTINE MATCAL
```

```
      REAL*8 A,AA,B,C,D,DD,S,X,Y
      DIMENSION AA(1),X(1),LL(9),MM(9),Y(9,9),S(1),D(1),
     1A(9,9),B(9,9),C(9,9)
C
C
      ENTRY INVERT(N,AA,X,KER,K)
C
C        THIS SUBROUTINE INVERTS THE MATRIX A AND LEAVES THE
C        RESULTS IN THE MATRIX X.   N AND K AREA THE ORDER OF THE MATRIX.
C        IF KER EQUALS 2 THEN A SINGULARITY HAS BEEN DETECTED.
C
      DO 1 I=1,N
      DO 1 J=1,N
      IND=(I-1)*K+J
    1 Y(I,J)=AA(IND)
C     KER=1
      N2=2*N
      CALL ARRAY(2,N,N,9,9,Y,Y)
      CALL DMINV(Y,N,DD,LL,MM)
      CALL ARRAY(1,N,N,9,9,Y,Y)
      IF(DD.EQ.0) KER=2
      DO 2 I=1,N
      DO 2 J=1,N
      IND=(I-1)*K+J
    2 X(IND)=Y(I,J)
      RETURN
C
C
      ENTRY ADDSUB(A,B,N,M,C,ISIGN)
C
C         THIS SUBROUTINE ADDS(ISIGN=1) OR SUBTRACTS(ISIGN=-1)
C         THE NXM MATRICES A AND B (A+B OR A-B) ,
C         STORING THE RESULT IN C.
C
      DO 125 I=1,N
      DO 125 J=1,M
  125 C(I,J)=A(I,J)+FLOAT(ISIGN)*B(I,J)
      RETURN
C
C
      ENTRY PROD(A,B,N,M,L,C)
C
C         THIS SUBROUTINE COMPUTES THE MATRIX PRODUCT AB
C         AND STORES THE RESULT IN C.  A IS NXM,B IS MXL, AND C IS NXL.
C
      DO 3 I=1,N
      DO 3 J=1,L
    3 C(I,J)=0.D0
```

```
      DO 126 I=1,N                                                      LENT2020
      DO 126 J=1,L                                                      LENT2030
      DO 126 K=1,M                                                      LENT2040
  126 C(I,J)=C(I,J)+A(I,K)*B(K,J)                                       LENT2050
      RETURN                                                            LENT2060
C                                                                       LENT2070
C                                                                       LENT2080
      ENTRY TRANS(A,N,M,C)                                              LENT2090
C                                                                       LENT2100
C        THIS SUBROUTINE TRANSPOSES THE  NXM MATRIX A                   LENT2110
C        AND STORES THE RESULT AS THE MXN MATRIX C.                     LENT2120
C                                                                       LENT2130
      DO 127 I=1,N                                                      LENT2140
      DO 127 J=1,M                                                      LENT2150
  127 C(J,I)=A(I,J)                                                     LENT2160
      RETURN                                                            LENT2170
      END                                                               LENT2180
      SUBROUTINE ARRAY(MODE,I,J,N,M,S,D)                                LENT2190
      DIMENSION S(1),D(1)                                               LENT2200
      REAL*8 S,D                                                        LENT2210
      NI=N-I                                                            LENT2220
      IF(MODE-1) 100,100,120                                            LENT2230
  100 IJ=I*J+1                                                          LENT2240
      NM=N*J+1                                                          LENT2250
      DO 110 K=1,J                                                      LENT2260
      NM=NM-NI                                                          LENT2270
      DO 110 L=1,I                                                      LENT2280
      IJ=IJ-1                                                           LENT2290
      NM=NM-1                                                           LENT2300
  110 D(NM)=S(IJ)                                                       LENT2310
      GO TO 140                                                         LENT2320
  120 IJ=0                                                              LENT2330
      NM=0                                                              LENT2340
      DO 130 K=1,J                                                      LENT2350
      DO 125 L=1,I                                                      LENT2360
      IJ=IJ+1                                                           LENT2370
      NM=NM+1                                                           LENT2380
  125 S(IJ)=D(NM)                                                       LENT2390
  130 NM=NM+NI                                                          LENT2400
  140 RETURN                                                            LENT2410
      END                                                               LENT2420
C                                                                       LENT2430
C                                                                       LENT2440
C                                                                       LENT2450
C                                                                       LENT2460
C                                                                       LENT2470
C                                                                       LENT2480
C                                                                       LENT2490
```

```
C
      SUBROUTINE DER
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/DERCOM/EKDTAB(112),VKDTAB(112),D2W,HT,IPLC,NNT
      COMMON/DIFCOM/HMX,HMN,H,EROR,X,XMX,VDR(16,16),N,NAC,IFL,IFX,ISTF
      DATA G/32.174049/
      VDR(12,1)=VDR(1,3)
      VDR(12,2)=VDR(1,4)
      GG=G*(1.D0-VDR(1,2)*(9.5897768D-08-VDR(1,2)*6.872006D-15))
      P=1.D0-6.875347D-06*VDR(1,2)*(1.D0-4.7948887D-08*VDR(1,2))
      PP=(-6.875347D-06*(1.D0-9.5897774D-08*VDR(1,2)))/P
      PL=-D2W*.076475137*(P**4.2561222)
      P2=DSQRT(P)
      VS=1116.4437*P2
      V21=VDR(1,3)*VDR(1,3)
      V22=VDR(1,4)*VDR(1,4)
      V2=V21+V22
      V=DSQRT(V2)
      VM=V/VS
      T  =EKDTAB(1)
      TP  =0.D0
      IF(VM.LE.VKDTAB(1))      GO TO 4
      T  =EKDTAB(NNT)
      IF(VM.GE.VKDTAB(NNT))      GO TO 4
      IPLC=NNT/2
    1 IF(VM.GT.VKDTAB(IPLC-1))      GO TO 2
      IPLC=IPLC-1
      GO TO 1
    2 IF(VM.LE.VKDTAB(IPLC))      GO TO 3
      IPLC=IPLC+1
      GO TO 2
    3 D1=(VKDTAB(IPLC-1)-VKDTAB(IPLC   ))*(VKDTAB(IPLC-1)-VKDTAB(IPLC+1))
      D2=(VKDTAB(IPLC   )-VKDTAB(IPLC-1))*(VKDTAB(IPLC   )-VKDTAB(IPLC+1))
      D3=(VKDTAB(IPLC+1)-VKDTAB(IPLC-1))*(VKDTAB(IPLC+1)-VKDTAB(IPLC   ))
      E1=(VM-VKDTAB(IPLC   ))*(VM-VKDTAB(IPLC+1))
      E2=(VM-VKDTAB(IPLC-1))*(VM-VKDTAB(IPLC+1))
      E3=(VM-VKDTAB(IPLC-1))*(VM-VKDTAB(IPLC   ))
      T   =E1/D1*EKDTAB(IPLC-1)+E2/D2*EKDTAB(IPLC)+E3/D3*EKDTAB(IPLC+1)
      E1=VM+VM-VKDTAB(IPLC   )-VKDTAB(IPLC+1)
      E2=VM+VM-VKDTAB(IPLC-1)-VKDTAB(IPLC+1)
      E3=VM+VM-VKDTAB(IPLC-1)-VKDTAB(IPLC   )
      TP   =E1/D1*EKDTAB(IPLC-1)+E2/D2*EKDTAB(IPLC)+E3/D3*EKDTAB(IPLC+1)
      TP=TP/VS
    4 VDR(12,3)=PL*V*VDR(1,3)*T
      VDR(12,4)=PL*V*VDR(1,4)*T-GG
      VDR(12,5)=VDR(1,11)
      VDR(12,6)=VDR(1,12)
      VDR(12,7)=VDR(1,13)
```

```
      VDR(12,8)=VDR(1,14)
      VDR(12,9)=VDR(1,15)
      VDR(12,10)=VDR(1,16)
      A44=4.2561222*T-.5D0*V*TP
      A32=PL*V*VDR(1,3)*PP*A44
      A33=PL*((V21+V2)/V*T+V21*TP)
      A34=PL*VDR(1,3)*VDR(1,4)*(T/V+TP)
      VDR(12,11)=A32*VDR(1, 8)+A33*VDR(1,11)+A34*VDR(1,14)
      VDR(12,12)=A32*VDR(1, 9)+A33*VDR(1,12)+A34*VDR(1,15)
      VDR(12,13)=A32*VDR(1,10)+A33*VDR(1,13)+A34*VDR(1,16)
      A42=PL*V*VDR(1,4)*PP*A44+G*(9.5897768D-08-VDR(1,2)*1.3944012D-14)
      A43=A34
      A44=PL*((V2+V22)/V*T+V22*TP)
      VDR(12,14)=A42*VDR(1, 8)+A43*VDR(1,11)+A44*VDR(1,14)
      VDR(12,15)=A42*VDR(1, 9)+A43*VDR(1,12)+A44*VDR(1,15)
      VDR(12,16)=A42*VDR(1,10)+A43*VDR(1,13)+A44*VDR(1,16)
      RETURN
      END
C
C
C
C
C
C
C
C
C
      SUBROUTINE OUT
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/DIFCOM/HMX,HMN,H,EROR,X,XMX,VDR(16,16),N,NAC,IFL,IFX,ISTF
      COMMON/DERCOM/EKDTAB(112),VKDTAB(112),D2W,HT,IPLC,NNT
      IF(X.EQ.0.D0)       JFL=1
      IF(JFL.GT.0)      GO TO 29
   27 G5=HT-VDR(1,2)
      G6=HT-VDR(2,2)
      G7=VDR(2,2)-VDR(1,2)
      G0=G5/G7
      G2=G0*G0
      G0=(G7-G5-G6)/G7*G2
      G2=G2*G6
      G3=G6/G7
      G3=G3*G3*G5
      XX=X-G0*H+G2/VDR(13,2)+G3/VDR(12,2)
      G5=XX-X
      G6=G5+H
      G0=G5/H
      G2=G0*G0
      G0=(H+G6+G6)/H*G2
```

```
      G2=G2*G6                                                          LENT3460
      G3=G6/H                                                           LENT3470
      G3=G3*G3*G5                                                       LENT3480
      G1=1.D0-G0                                                        LENT3490
      DO 2 I=1,10                                                       LENT3500
    2 VDR(1,I)=G0*VDR(2,I)+G1*VDR(1,I)+G2*VDR(13,I)+G3*VDR(12,I)        LENT3510
      X=XX                                                              LENT3520
      IFL=-1                                                            LENT3530
      VDR(4,1)=-VDR(1, 8)/VDR(1,4)                                      LENT3540
      VDR(4,2)=-VDR(1, 9)/VDR(1,4)                                      LENT3550
      VDR(4,3)=-VDR(1,10)/VDR(1,4)                                      LENT3560
      G3=VDR(1,3)                                                       LENT3570
      VDR(4,4)=VDR(1,5)+G3*VDR(4,1)                                     LENT3580
      VDR(4,5)=VDR(1,6)+G3*VDR(4,2)                                     LENT3590
      VDR(4,6)=VDR(1,7)+G3*VDR(4,3)                                     LENT3600
      GO TO 3                                                           LENT3610
   29 IF(VDR(1,2).LE.HT)      GO TO 27                                  LENT3620
      YP=VDR(1,2)+H*(VDR(12,2)-16.087D0*H)-HT                           LENT3630
      IF(YP.GT.0.D0)      GO TO 3                                       LENT3640
      YP=VDR(1,2)-HT                                                    LENT3650
      H=-(YP+YP)/(VDR(12,2)-DSQRT(VDR(12,2)*VDR(12,2)-2.D0*YP*VDR(12,4))LENT3660
     1)                                                                 LENT3670
      JFL=0                                                             LENT3680
    3 IF(X.GE.XMX)      IFL=-1                                          LENT3690
      RETURN                                                            LENT3700
      END                                                               LENT3710
C                                                                       LENT3720
C                                                                       LENT3730
C                                                                       LENT3740
C                                                                       LENT3750
C                                                                       LENT3760
C                                                                       LENT3770
C                                                                       LENT3780
C                                                                       LENT3790
      SUBROUTINE STIFF(SUBA,SUBB)                                       LENT3800
      IMPLICIT REAL*8 (A-H,O-Z)                                         LENT3810
      COMMON/DIFCOM/HMX,HMN,H,EROR,X,XMX,VDR(16,16),N,NAC,IFL,IFX,ISTF  LENT3820
      IF(NAC.LE.0)      NAC=N                                           LENT3830
      IFL=-2                                                            LENT3840
    1 CALL SUBA                                                         LENT3850
      CALL SUBB                                                         LENT3860
      IF(IFL+1)2,22,3                                                   LENT3870
    2 IST=0                                                             LENT3880
      FIP=EROR                                                          LENT3890
      EPL=.005D0*EROR                                                   LENT3900
      DO 200 I=1,N                                                      LENT3910
  200 VDR(11,I)=DABS(VDR(1,I))                                          LENT3920
      IFL=0                                                             LENT3930
```

```
    3 H2=H+H                                                            LENT3940
      IF(F1P.GT.EPL.OR.IFX.GT.0.OR.DABS(H2).GT.HMX)     GO TO 4         LENT3950
      JST=IST/2                                                         LENT3960
      KST=IST-JST-JST                                                   LENT3970
      IF(KST.GT.0)     GO TO 4                                          LENT3980
      IST=JST                                                           LENT3990
      H=H2                                                              LENT4000
    4 IT=1                                                              LENT4010
      XS=X                                                              LENT4020
      H2=.5D0*H                                                         LENT4030
      DO 5 I=1,N                                                        LENT4040
      VDR(2,I)=VDR(1,I)                                                 LENT4050
      VDR(10,I)=VDR(1,I)                                                LENT4060
      VDR(13,I)=VDR(12,I)                                               LENT4070
    5 VDR(16,I)=VDR(12,I)                                               LENT4080
    6 X=X+H2                                                            LENT4090
      DO 7 I=1,N                                                        LENT4100
    7 VDR(1,I)=VDR(10,I)+H2*VDR(16,I)                                   LENT4110
      CALL SUBA                                                         LENT4120
      DO 8 I=1,N                                                        LENT4130
      VDR(3,I)=VDR(1,I)                                                 LENT4140
      VDR(14,I)=VDR(12,I)                                               LENT4150
    8 VDR(1,I)=VDR(10,I)+H2*VDR(12,I)                                   LENT4160
      CALL SUBA                                                         LENT4170
      DO 10 I=1,N                                                       LENT4180
      VDR(4,I)=VDR(1,I)                                                 LENT4190
      VDR(15,I)=VDR(12,I)                                               LENT4200
      IF(ISTF.EQ.0)     GO TO 9                                         LENT4210
      P=VDR(4,I)-VDR(3,I)                                               LENT4220
      IF(P.EQ.0.D0)     GO TO 9                                         LENT4230
      P=-(VDR(15,I)-VDR(14,I))/P                                        LENT4240
      PH=P*H                                                            LENT4250
      IF(PH.LT..5D0)     GO TO 9                                        LENT4260
      VDR(5,I)=P                                                        LENT4270
      P=DEXP(-PH)                                                       LENT4280
      VDR(6,I)=(1.D0-P)/PH                                              LENT4290
      VDR(7,I)=(1.D0-VDR(6,I))/PH                                       LENT4300
      VDR(8,I)=(.5D0-VDR(7,I))/PH                                       LENT4310
      VDR(1,I)=VDR(10,I)+H*((VDR(15,I)+VDR(15,I))*VDR(7,I)+VDR(13,I)*(VDLENT4320
     1R(6,I)-VDR(7,I)-VDR(7,I))+VDR(14,I)*VDR(7,I)*PH)                  LENT4330
      GO TO 10                                                          LENT4340
    9 VDR(5,I)=0.D0                                                     LENT4350
      VDR(1,I)=VDR(10,I)+H*VDR(15,I)                                    LENT4360
   10 CONTINUE                                                          LENT4370
      X=X+H2                                                            LENT4380
      CALL SUBA                                                         LENT4390
      DO 12 I=1,N                                                       LENT4400
      P=VDR(5,I)                                                        LENT4410
```

```
      IF(ISTF.EQ.0.OR.P.EQ.0.D0)      GO TO 11                           LENT4420
      PH=VDR(6,I)*VDR(16,I)                                               LENT4430
      F1P=VDR(16,I)+P*VDR(10,I)                                           LENT4440
      F2P=VDR(14,I)+P*VDR( 3,I)                                           LENT4450
      F3P=VDR(15,I)+P*VDR( 4,I)                                           LENT4460
      F4P=VDR(12,I)+P*VDR( 1,I)                                           LENT4470
      VDR(1,I)=VDR(10,I)+H*(PH+VDR(7,I)*(-F1P-F1P-F1P+F2P+F2P+F3P+F3P-F4LENT4480
     1P)+4.D0*(F1P-F2P-F3P+F4P)*VDR(8,I))                                LENT4490
      GO TO 12                                                            LENT4500
   11 VDR(1,I)=VDR(10,I)+H/6.D0*(VDR(16,I)+VDR(14,I)+VDR(14,I)+VDR(15,I)LENT4510
     1+VDR(15,I)+VDR(12,I))                                              LENT4520
   12 CONTINUE                                                            LENT4530
      IF(IFX.GT.0)      GO TO 1                                           LENT4540
      IF(IT-2)13,15,17                                                    LENT4550
   13 DO 14 I=1,N                                                         LENT4560
   14 VDR(9,I)=VDR(1,I)                                                   LENT4570
      X=XS                                                                LENT4580
      IT=2                                                                LENT4590
      H=H2                                                                LENT4600
      H2=.5D0*H                                                           LENT4610
      GO TO 6                                                             LENT4620
   15 CALL SUBA                                                           LENT4630
      DO 16 I=1,N                                                         LENT4640
      VDR(10,I)=VDR( 1,I)                                                 LENT4650
   16 VDR(16,I)=VDR(12,I)                                                 LENT4660
      IT=3                                                                LENT4670
      GO TO 6                                                             LENT4680
   17 F1P=0.D0                                                            LENT4690
      DO 18 I=1,N                                                         LENT4700
      F2P=(VDR(1,I)-VDR(9,I))/15.D0                                       LENT4710
      IF(I.GT.NAC)      GO TO 18                                          LENT4720
      F3P=DABS(F2P)                                                       LENT4730
      F4P=DABS(VDR(1,I))                                                  LENT4740
      P=VDR(11,I)                                                         LENT4750
      IF(F4P.GT.P)      P=F4P                                             LENT4760
      IF(P.NE.0.D0)      F3P=F3P/P                                        LENT4770
      IF(F3P.GT.EROR)      GO TO 19                                       LENT4780
      IF(F3P.GT.F1P)      F1P=F3P                                         LENT4790
   18 VDR(1,I)=VDR(1,I)+F2P                                               LENT4800
      IST=IST+1                                                           LENT4810
      H=H+H                                                               LENT4820
      DO 180 I=1,N                                                        LENT4830
      F3P=DABS(VDR(1,I))                                                  LENT4840
  180 IF(F3P.GT.VDR(11,I))      VDR(11,I)=F3P                             LENT4850
      GO TO 1                                                             LENT4860
   19 H=H2                                                                LENT4870
      IF(DABS(H2).GT.HMN)      GO TO 20                                   LENT4880
      IFL=-1                                                              LENT4890
```

```
      GO TO 1                                                           LENT4900
   20 X=XS                                                              LENT4910
      H2=.5D0*H                                                         LENT4920
      IT=2                                                              LENT4930
      DO 21 I=1,N                                                       LENT4940
      VDR( 9,I)=VDR(10,I)                                               LENT4950
      VDR(10,I)=VDR( 2,I)                                               LENT4960
   21 VDR(16,I)=VDR(13,I)                                               LENT4970
      IST=IST+IST                                                       LENT4980
      GO TO 6                                                           LENT4990
   22 RETURN                                                            LENT5000
      END                                                               LENT5010
C                                                                       LENT5020
C                                                                       LENT5030
C                                                                       LENT5040
C                                                                       LENT5050
C                                                                       LENT5060
C                                                                       LENT5070
C                                                                       LENT5080
C                                                                       LENT5090
      SUBROUTINE MATMAD(A,Z,X,Y)                                        LENT5100
      REAL*8 A(3,3),X(3),Y(3),Z(3)                                      LENT5110
      IGO=1                                                             LENT5120
      GO TO 1                                                           LENT5130
      ENTRY MATMLT(A,X,Y)                                               LENT5140
      IGO=0                                                             LENT5150
    1 DO 2 I=1,3                                                        LENT5160
      Y(I)=0.D0                                                         LENT5170
      IF(IGO.NE.0)      Y(I)=Z(I)                                       LENT5180
      DO 2 J=1,3                                                        LENT5190
    2 Y(I)=Y(I)+A(I,J)*X(J)                                             LENT5200
      RETURN                                                            LENT5210
      END                                                               LENT5220
C                                                                       LENT5230
C                                                                       LENT5240
C                                                                       LENT5250
C                                                                       LENT5260
C                                                                       LENT5270
C                                                                       LENT5280
C                                                                       LENT5290
C                                                                       LENT5300
      BLOCK DATA                                                        LENT5310
C                                                                       LENT5320
C----SET ENTIRE COMMON BLOCK TO ZERO                                    LENT5330
      IMPLICIT REAL*8(Z)                                                LENT5340
      COMMON/AIRCOM/Z1(25)                                              LENT5350
      COMMON/RADCOM/Z2(37)                                              LENT5360
      COMMON/DERCOM/Z3(228)                                             LENT5370
```

```
      COMMON/DIFCOM/Z4(267)
      DATA Z1/25*0.D0/
      DATA Z2/37*0.D0/
      DATA Z3/228*0.D0/
      DATA Z4/267*0.D0/
      END
C
C
C
C
C
C
C
C
//GO.SYSIN DD *

 15.0     0.1        0.1

-50000.    20000.     10000.
 500.
 0.0        2.0       0.125     0.125     0.        0.
 10.        5.        6.        0.        1.         1.
 1.         1.        1.        1.        9999.
 X RESIDUAL VS. T

 Y RESIDUAL VS. T

 Z RESIDUAL VS. T

 RADIAL RESIDUAL VS. T

 X6(1) VS. T

 X6(2) VS. T

 X6(2) VS. X6(1)

 LATERAL ERROR (XE) VS. T

 LATERAL ERROR (EXPANDED SCALE) VS. T

 LATERAL ERROR RATE (DXE) VS. T

 PHD,PHD1, AND PHD2 VS. T

 ROLL ANGLE(COMMAND,ACTUAL,&EST) VS. T
```

```
BLANK-STITLE                                                            LENT5850
                                                                        LENT5860
X ACCELERATION ESTIMATE(XDD1(1)) VS. T                                  LENT5870
                                                                        LENT5880
Y ACCELERATION ESTIMATE(XDD1(2)) VS. T                                  LENT5890
                                                                        LENT5900
X ACCELERATION VS. Y ACCELERATION                                       LENT5910
                                                                        LENT5920
BLANK-WTITLE                                                            LENT5930
                                                                        LENT5940
BLANK-XTITLE                                                            LENT5950
                                                                        LENT5960
BLANK-YTITLE                                                            LENT5970
                                                                        LENT5980
BLANK-ZTITLE                                                            LENT5990
                                                                        LENT6000
 0.        0.        1.0       4.        23.6      144                  LENT6010
 7   7112 61 33 62 33 64 38 76 38144 85148 51                           LENT6020
0.      .0428 .81     .04285.82     .0429 .83     .04295.84     .0434 .85     .0441 LENT6030
.86     .0451 .87     .0464 .88     .0479 .89     .0499 .9      .0527 .905    .0543 LENT6040
.91     .056  .915    .0577 .92     .0595 .925    .0613 .93     .0631 .935    .065  LENT6050
.94     .0671 .945    .0691 .95     .0712 .955    .0734 .96     .0756 .965    .0779 LENT6060
.97     .0803 .975    .0827 .98     .0852 .985    .0878 .99     .0904 .995    .0931 LENT6070
1.      .0958 1.005   .0986 1.01    .1015 1.015   .1044 1.02    .1074 1.025   .1104 LENT6080
1.03    .1135 1.035   .1165 1.04    .1193 1.045   .1218 1.05    .1238 1.055   .1253 LENT6090
1.06    .1263 1.065   .127  1.07    .1276 1.075   .1283 1.08    .129  1.085   .1296 LENT6100
1.09    .1301 1.095   .1304 1.1     .1307 1.11    .1315 1.12    .132  1.13    .1323 LENT6110
1.14    .1325 1.15    .1326 1.16    .1328 1.17    .1331 1.18    .1334 1.19    .1336 LENT6120
1.2     .1339 1.21    .1342 1.22    .1344 1.23    .1347 1.24    .1349 1.25    .1352 LENT6130
1.26    .1355 1.27    .1357 1.28    .136  1.29    .1362 1.3     .1365 1.31    .1368 LENT6140
1.32    .137  1.33    .1373 1.34    .1376 1.35    .1378 1.36    .1381 1.37    .1384 LENT6150
1.38    .1386 1.39    .1389 1.4     .1391 1.41    .1393 1.42    .1394 1.43    .1395 LENT6160
1.44    .1395 1.45    .1394 1.46    .1393 1.47    .1392 1.48    .1391 1.49    .1389 LENT6170
1.5     .1387 1.51    .1384 1.52    .1381 1.53    .1379 1.54    .1376 1.55    .1373 LENT6180
1.56    .137  1.57    .1367 1.58    .1364 1.59    .1361 1.6     .1358 1.61    .1355 LENT6190
1.62    .1353 1.63    .135  1.64    .1347 1.65    .1345 1.66    .1343 1.67    .1342 LENT6200
1.68    .1341 1.69    .13395 1.7    .134  2.5     .13405                            LENT6210
0.      .0871 .44     .08715.46     .0878 .48     .0885 .5      .0891 .52     .0899 LENT6220
.54     .0907 .56     .0915 .58     .0925 .6      .0933 .62     .0943 .64     .0953 LENT6230
.66     .0962 .68     .0974 .7      .0985 .72     .0996 .74     .1007 .76     .1019 LENT6240
.9      .1106 .92     .1119 .94     .1133 .96     .1147 .98     .116  1.      .1174 LENT6250
1.02    .1188 1.04    .1202 1.06    .1215                                       LENT6260
.78     .103  .8      .1043 .82     .1054 .84     .1067 .86     .1089 .88     .1093 LENT6270
0.      .0549 .44     .05495.46     .0553 .48     .0556 .5      .0561 .52     .0565 LENT6280
.54     .0571 .56     .0576 .58     .0582 .6      .0588 .62     .0595 .64     .0601 LENT6290
.66     .0608 .68     .0615 .7      .062  .72     .0629 .74     .0637 .76     .0644 LENT6300
.78     .0651 .8      .0658 .82     .0665 .84     .0672 .86     .0679 .88     .0687 LENT6310
.9      .0697 .92     .0708 .94     .0723 .96     .074  .98     .0762 1.      .0785 LENT6320
```

1.02	.0811	1.04	.084	1.06	.0868							LENT6330
0.	.1646	.4	.1645	.6	.163	.7	.1618	.8	.1609	.9	.1618	LENT6340
.91	.1623	.92	.1629	.93	.164	.94	.1654	.95	.1674	.96	.1703	LENT6350
.97	.1744	.98	.1791	.99	.1863	1.	.1944	1.01	.2009	1.02	.2066	LENT6360
1.03	.211	1.04	.214	1.05	.2161	1.06	.2176	1.07	.2187	1.08	.2195	LENT6370
1.09	.2201	1.1	.2207	1.12	.2213	1.14	.2219	1.16	.2223	1.18	.2227	LENT6380
1.2	.223	1.3	.2223	1.4	.2188	1.5	.2144	1.6	.2097	1.7	.2066	LENT6390
1.8	.2046	2.5	.2042									LENT6400
0.	.1619	.7	.1619	.75	.1619	.8	.1619	.85	.1619	.9	.162	LENT6410
.91	.162	.92	.1621	.93	.1621	.94	.1623	.95	.1624	.96	.1627	LENT6420
.97	.1634	.98	.165	.99	.1698	1.	.1837	1.01	.1953	1.02	.1989	LENT6430
1.03	.2002	1.04	.2011	1.05	.2017	1.06	.2022	1.07	.2023	1.08	.2026	LENT6440
1.1	.2029	1.12	.203	1.14	.2029	1.16	.2028	1.18	.2026	1.2	.202	LENT6450
1.3	.199	1.4	.1951	1.5	.1907	1.6	.1875	1.7	.1858	1.8	.1853	LENT6460
1.9	.1653	2.	.1853									LENT6470
0.	.1048	.5	.1049	.52	.1052	.54	.1056	.56	.106	.58	.1065	LENT6480
.6	.1072	.62	.1079	.64	.1088	.66	.1098	.68	.111	.7	.1125	LENT6490
.72	.1142	.74	.1163	.76	.1186	.78	.1215	.8	.1243	.82	.1283	LENT6500
.84	.1329	.86	.1386	.88	.1453	.89	.1495	.9	.1543	.91	.1596	LENT6510
.92	.1658	.93	.1733	.94	.1818	.95	.1916	.96	.2024	.97	.2136	LENT6520
.98	.2258	.99	.2386	1.	.2523	1.01	.2655	1.02	.2777	1.03	.2878	LENT6530
1.04	.2965	1.05	.3043	1.06	.3104	1.07	.3157	1.08	.3204	1.09	.3248	LENT6540
1.1	.3286	1.11	.3323	1.12	.3353	1.13	.3379	1.14	.3414	1.15	.3442	LENT6550
1.16	.3467	1.17	.3495	1.18	.3521	1.19	.3545	1.2	.3567	1.21	.3589	LENT6560
1.22	.3607	1.23	.3627	1.24	.3644	1.26	.368	1.28	.3713	1.3	.3745	LENT6570
1.32	.3771	1.34	.3798	1.36	.3824	1.38	.3847	1.4	.3866	1.42	.3884	LENT6580
1.44	.39	1.46	.3912	1.48	.3923	1.5	.3931	1.52	.3939	1.54	.3945	LENT6590
1.56	.3951	1.58	.3954	1.6	.3958	1.65	.3966	1.7	.397	1.75	.3964	LENT6600
1.8	.3954	1.85	.3941	1.9	.3927	1.95	.3913	2.	.3901	2.1	.3896	LENT6610
3.	.38965											LENT6620
0.	.0531	.8	.05315	.82	.0532	.83	.0534	.84	.0541	.85	.0554	LENT6630
.86	.0571	.87	.0593	.88	.062	.89	.0651	.9	.0689	.91	.0732	LENT6640
.92	.0781	.93	.0845	.94	.0924	.95	.1018	.96	.1135	.97	.128	LENT6650
.98	.1431	.99	.1558	1.	.1665	1.01	.1757	1.02	.183	1.03	.1885	LENT6660
1.04	.1924	1.05	.1962	1.06	.1986	1.07	.2009	1.08	.2028	1.09	.2043	LENT6670
1.1	.2054	1.12	.2062	1.14	.2065	1.16	.2065	1.18	.2064	1.2	.2063	LENT6680
1.22	.2063	1.24	.2062	1.26	.2062	1.28	.2061	1.3	.206	1.4	.2055	LENT6690
1.5	.2043	1.6	.2026	1.7	.2009	1.8	.1993	1.9	.1976	2.	.196	LENT6700
2.2	.1926	2.5	.1876	3.	.1793							LENT6710

```
C                                                                          LENT0010
C                                                                          LENT0020
C**********AN/TPQ-27 SIMULATION PROGRAM FOR COARSE GUIDANCE*************LENT0030
C                                                                          LENT0040
C                    ROBERT E. LENTZ                                       LENT0050
C                                                                          LENT0060
C                                                                          LENT0070
      IMPLICIT REAL*8 (A-H,O-Z)                                            LENT0080
C                                                                          LENT0090
      COMMON/AIRCOM/X3(3),XD3(3),WT(3),PHB,TB,DT,PHC,ITH                   LENT0100
C                                                                          LENT0110
      COMMON/RADCOM/SIG(6),BIS(3),X1(3),XD1(3),PS1,PH1,WR(3),              LENT0120
     1              DTRAD,SIGW(3),IU,NWLD                                  LENT0130
C                                                                          LENT0140
C----DIMENSION STORAGE FOR PLOTTING ARRAYS AND ARRAY TITLES                LENT0150
      REAL*4 XXA(200),XXB(200),XXC(200),XXD(200),XXE(200)                  LENT0160
      REAL*4 YYA(200),YYB(200),YYC(200),YYD(200),YYE(200)                  LENT0170
      REAL*8 ATITLE(12),BTITLE(12),CTITLE(12),DTITLE(12),ETITLE(12)        LENT0180
C                                                                          LENT0190
      DIMENSION XWP(5),YWP(5),THETA(5),H(4),VGH(4),PLENTH(4),              LENT0200
     1RANGE(4),AZ(4),TLEG(4),DEL(3),X2(3),DELNM(3),DX(3),DXD(3).WH(3),     LENT0210
     2VGX(4),VGY(4),X1NM(3),X2NM(3),X3NM(3),XWPNM(5),YWPNM(5)              LENT0220
C                                                                          LENT0230
      DATA RAD,G/57.295779513082321,32.174049/                             LENT0240
      DATA PI,FEET,EPS/3.141592654,6076.1,0.0001/                          LENT0250
      DATA G1,G2,PHILIM/4.0,4.0,30.0/                                      LENT0260
      DATA ITABA,ITAB1,IAVG.IEND.ITURN/4*0,-1/                             LENT0270
      DATA T,TTURN1,TINTPN,TT21,TO21,SUMEST,SUMETR/7*0.D0/                 LENT0280
      DATA HDE,XINT,YINT,TLEGI,XINTNM,YINTNM,TSTOP/6*0.D0,1.D06/           LENT0290
      DATA PHD1,PHD2,LNCLIM/2*0.0,72/                                      LENT0300
C                                                                          LENT0310
      NWLD=-1                                                              LENT0320
      DEG=1./RAD                                                           LENT0330
      TAN=DEG*PHILIM                                                       LENT0340
      HERMIN=5.0                                                           LENT0350
C                                                                          LENT0360
C----DT IS CONTROL TIME INTERVAL                                           LENT0370
C    DTRAD IS RADAR SAMPLING INTERVAL                                      LENT0380
      IDTRAD=6                                                             LENT0390
      DTRAD=FLOAT(IDTRAD)                                                  LENT0400
      DT=1.0                                                               LENT0410
      DTCON=DT                                                             LENT0420
      IU=13521465                                                          LENT0430
C                                                                          LENT0440
C----RADAR MEASUREMENT STANDARD DEVIATIONS AND RANDOM FORCING VALUES       LENT0450
      SIG(1)=900.                                                          LENT0460
      SIG(2)=0.4                                                           LENT0470
      SIG(3)=0.4                                                           LENT0480
```

```
C
      SIGW(1)=0.0
      SIGW(2)=0.0
      SIGW(3)=0.0
C
C----TB IS TRUE AIRCRAFT ROLL RESPONSE TIME CONSTANT
C    TAUH IS THE ESTIMATED AIRCRAFT ROLL RESPONSE TIME CONSTANT
      TB=3.3
      TAUH=TB
C
C----READ INPUT PARAMETERS
      READ(5,100) NLEG,RANGEI,AZI,VA,VAH,VW,VWH,THETW,THETWH
      READ(5,110) (PLENTH(I),THETA(I),I=1,NLEG)
      READ(5,111) HTG,DELNM
      READ(5,112) ATITLE
C
C----COMPUTE TRUE WIND (WT) AND ESTIMATED WIND (WH) COMPONENTS
      WT(1)=VW*DSIN(THETW*DEG)
      WT(2)=VW*DCOS(THETW*DEG)
      WT(3)=0.0
      WH(1)=VWH*DSIN(THETWH*DEG)
      WH(2)=VWH*DCOS(THETWH*DEG)
      WH(3)=0.0
      DO 2 I=1,3
    2 WP(I)=WH(I)
C
C----HTG IS INITIAL TRUE GROUND HEADING
C    HTA IS INITIAL TRUE AIR HEADING
C    VTG IS INITIAL TRUE GROUND SPEED
C    VA IS THE TRUE AIR SPEED THROUGHOUT PROBLEM
      U1=VW*DSIN((THETW-HTG)*DEG)/VA
      HTA=HTG-RAD*DARSIN(U1)
      IF(HTA.LT.0.) HTA=HTA+360.
      IF(HTA.GT.360.) HTA=HTA-360.
      VTG=VW*DCOS((THETW-HTG)*DEG)+VA*DCOS((HTA-HTG)*DEG)
C
C----COMPUTE INITIAL TRUE A/C VELOCITY
      XD3(1)=VTG*DSIN(HTG*DEG)
      XD3(2)=VTG*DCOS(HTG*DEG)
      XD3(3)=0.
C
C----COMPUTE LEG PARAMETERS FOR MISSION DATA TABLE
C    XWP,YWP ARE COORDINATES OF LEG START/STOP POINTS
C    RANGE IS AVG DISTANCE TO LEG FROM RADAR; AZ IS AVG ANGLE TO LEG
C    H IS DESIRED AIR SPEED HEADING ALONG LEG
C    VGH IS DESIRED GROUND SPEED ALONG LEG
C    VGX,VGY ARE COMPONENTS OF VGH
      XWP(1)=RANGEI*FEET*DSIN(AZI*DEG)
```

```
      YWP(1)=RANGEI*FEET*DCOS(AZI*DEG)                                  LENT0970
      XWPNM(1)=XWP(1)/FEET                                              LENT0980
      YWPNM(1)=YWP(1)/FEET                                              LENT0990
      XMAX=XWPNM(1)                                                     LENT1000
      XMIN=XWPNM(1)                                                     LENT1010
      YMAX=YWPNM(1)                                                     LENT1020
      YMIN=YWPNM(1)                                                     LENT1030
C                                                                       LENT1040
C----COMPUTE INITIAL TRUE A/C POSITION                                  LENT1050
      DO 9 KK=1,3                                                       LENT1060
    9 DEL(KK)=DELNM(KK)*FEET                                            LENT1070
      X3(1)=XWP(1)+DEL(1)                                               LENT1080
      X3(2)=YWP(1)+DEL(2)                                               LENT1090
      X3(3)=DEL(3)                                                      LENT1100
      DO 10 KK=1,3                                                      LENT1110
   10 X3NM(KK)=X3(KK)/FEET                                              LENT1120
C                                                                       LENT1130
C----PRINT OUT SIMULATION INITIAL CONDITIONS AND PARAMETERS             LENT1140
      PRINT 1000,VW,THETW,WT(1),WT(2),VWH,THETWH,WH(1),WH(2),           LENT1150
     1           SIG(1),SIG(2),SIG(3),BIS,SIG(4),SIG(5),SIG(6),         LENT1160
     2           SIGW,DTRAD,DELNM,VTG,HTG,XD3(1),XD3(2),VA,HTA,TB       LENT1170
      PRINT 1001,NLEG,RANGEI,AZI,XWPNM(1),YWPNM(1),DTCON,PHILIM,        LENT1180
     1           G1,G2,HERMIN                                           LENT1190
C                                                                       LENT1200
C                                                                       LENT1210
C----COMPUTE MISSION DATA TABLE                                         LENT1220
      DO 120   I=1,NLEG                                                 LENT1230
      XWP(I+1)=XWP(I)+PLENTH(I)*FEET*DSIN(DEG*THETA(I))                 LENT1240
      YWP(I+1)=YWP(I)+PLENTH(I)*FEET*DCOS(DEG*THETA(I))                 LENT1250
      XWPNM(I+1)=XWP(I+1)/FEET                                          LENT1260
      YWPNM(I+1)=YWP(I+1)/FEET                                          LENT1270
      RANGE(I)=0.5*DSQRT((XWP(I)+XWP(I+1))**2+(YWP(I)+YWP(I+1))**2)/FEETLENT1280
      AZ(I)=RAD*DATAN2(XWP(I)+XWP(I+1),YWP(I)+YWP(I+1))                 LENT1290
      U1=VWH*DSIN((THETWH-THETA(I))*DEG)/VAH                            LENT1300
      H(I)=THETA(I)-RAD*DARSIN(U1)                                      LENT1310
      IF(H(I).LT.0.) H(I)=H(I)+360.                                     LENT1320
      IF(H(I).GE.360.) H(I)=H(I)-360.                                   LENT1330
      VGH(I)=VAH*DCOS(DEG*(H(I)-THETA(I)))                              LENT1340
     1       +VWH*DCOS(DEG*(THETWH-THETA(I)))                           LENT1350
      TLEG(I)=(PLENTH(I)*FEET)/VGH(I)                                   LENT1360
      VGX(I)=VGH(I)*DSIN(THETA(I)*DEG)                                  LENT1370
  120 VGY(I)=VGH(I)*DCOS(THETA(I)*DEG)                                  LENT1380
C                                                                       LENT1390
C----SET UP X2 VECTOR WHICH WILL PLOT AS DESIRED TRACK                  LENT1400
      DO 123 I=1.NLEG                                                   LENT1410
      ITABA=ITABA+1                                                     LENT1420
      X2NM(1)=XWPNM(I)                                                  LENT1430
      X2NM(2)=YWPNM(I)                                                  LENT1440
```

```
      XXC(ITABA)=X2NM(1)
      YYC(ITABA)=X2NM(2)
      DT2PLT=TLEG(I)/DTRAD
      IDT2=DT2PLT
      DO 121 J=1,IDT2
      ITABA=ITABA+1
      X2NM(1)=X2NM(1)+(DTRAD*VGX(I))/FEET
      X2NM(2)=X2NM(2)+(DTRAD*VGY(I))/FEET
      IF(X2NM(1).GT.XMAX) XMAX=X2NM(1)
      IF(X2NM(1).LT.XMIN) XMIN=X2NM(1)
      IF(X2NM(2).LT.YMIN) YMIN=X2NM(2)
      IF(X2NM(2).GT.YMAX) YMAX=X2NM(2)
      XXC(ITABA)=X2NM(1)
  121 YYC(ITABA)=X2NM(2)
  123 CONTINUE
C
C----SET UP PROBLEM INITIAL CONDITIONS
      I=1
      CALCST=2.0*DTRAD+1.0-EPS
      CALCS1=CALCST+1.0
      HSTART=H(1)
      HNEXT=H(2)
C
      PRINT 1100,I,XWPNM(I),YWPNM(I),XWPNM(I+1),YWPNM(I+1),PLENTH(I),
     1          THETA(I),VGH(I),H(I),RANGE(I),AZ(I),TLEG(I)
      PRINT 1101
      PRINT 1200
      LNC=3
C
C***********************************************************************
C
C----BEGIN MAIN PROCESSING STREAM
C
C----THE ARCRFT SUBROUTINE CALCULATES TRUE NEW POSITION AND
C    VELOCITY OF THE AIRCRAFT IN CARTESIAN COORDINATES
  191 CALL ARCRFT
      DO 192 KK=1,3
  192 X3NM(KK)=X3(KK)/FEET
C
C----THE RADAR SUBROUTINE CALCULATES THE ESTIMATED POSITION
C    AND VELOCITY OF THE AIRCRAFT IN CARTESIAN COORDINATES
      CALL RADAR6
      DO 203  KK=1,3
  203 X1NM(KK)=X1(KK)/FEET
C
C----CALCULATE HEADINGS AND ERRORS FOR USE IN CONTROL CALCULATION
C    HTG IS TRUE GROUNG HEADING
C    HEG IS THE ESTIMATED GROUNG HEADING
```

```
C     HTA IS TRUE AIR SPEED HEADING
C     HEA IS THE ESTIMATED AIR SPEED HEADING
      HTG=DATAN2(XD3(1),XD3(2))*RAD
      HTA=DATAN2((XD3(1)-WT(1)),(XD3(2)-WT(2)))*RAD
      IF(T.LT.(DTRAD-EPS)) GO TO 205
      HEA=DATAN2((XD1(1)-WH(1)),(XD1(2)-WH(2)))*RAD
      HEG=DATAN2(XD1(1),XD1(2))*RAD
      VEG=DSQRT(XD1(1)**2+XD1(2)**2)
      GO TO 206
 205  HEA=0.D0
      HEG=0.D0
C
 206  IF(HTG.GT.360.) HTG=HTG-360.
      IF(HEG.GT.360.) HEG=HEG-360.
      IF(HTA.GT.360.) HTA=HTA-360.
      IF(HEA.GT.360.) HEA=HEA-360.
      IF(HTG.LT.   0.) HTG=HTG+360.
      IF(HEG.LT.   0.) HEG=HEG+360.
      IF(HTA.LT.   0.) HTA=HTA+360.
      IF(HEA.LT.   0.) HEA=HEA+360.
C
C----ETRUE IS THE TRUE A/C DISPLACEMENT FROM DESIRED TRACK
C    EEST IS THE ESTIMATED DISPLACEMENT OF A/C FROM THE
C    DESIRED TRACK
      ETRUE=(X3(1)-XWP(I))*DCOS(THETA(I)*DEG)/FEET
     1      -(X3(2)-YWP(I))*DSIN(THETA(I)*DEG)/FEET
      EEST=(X1(1)-XWP(I))*DCOS(THETA(I)*DEG)/FEET
     1      -(X1(2)-YWP(I))*DSIN(THETA(I)*DEG)/FEET
      IF(ITURN.NE.0) GO TO 2058
      IF(IEND.EQ.1) GO TO 2058
      ETRUE1=(X3(1)-XWP(I+1))*DCOS(THETA(I+1)*DEG)/FEET
     1      -(X3(2)-YWP(I+1))*DSIN(THETA(I+1)*DEG)/FEET
      EEST1=(X1(1)-XWP(I+1))*DCOS(THETA(I+1)*DEG)/FEET
     1      -(X1(2)-YWP(I+1))*DSIN(THETA(I+1)*DEG)/FEET
      IF(DABS(ETRUE1).LT.DABS(ETRUE)) ETRUE=ETRUE1
      IF(DABS(EEST1).LT.DABS(EEST)) EEST=EEST1
C
C----COMPUTE OFF TRACK ERROR RMS SUMS
2053  IAVG=IAVG+1
      SUMEST=SUMEST+EEST**2
      SUMETR=SUMETR+ETRUE**2
C
C----COMPUTE PLOTTING LIMITS FOR SCALING
      IF(MOD(ITH,IDTRAD).NE.0) GO TO 2059
      ITAB1=ITAB1+1
      IF(X1NM(1).GT.XMAX) XMAX=X1NM(1)
      IF(X3NM(1).GT.XMAX) XMAX=X3NM(1)
      IF(X1NM(1).LT.XMIN) XMIN=X1NM(1)
```

```
      IF(X3NM(1).LT.XMIN) XMIN=X3NM(1)
      IF(X1NM(2).LT.YMIN) YMIN=X1NM(2)
      IF(X3NM(2).LT.YMIN) YMIN=X3NM(2)
      IF(X1NM(2).GT.YMAX) YMAX=X1NM(2)
      IF(X3NM(2).GT.YMAX) YMAX=X3NM(2)
      XXA(ITAB1)=X1NM(1)
      YYA(ITAB1)=X1NM(2)
      XXB(ITAB1)=X3NM(1)
      YYB(ITAB1)=X3NM(2)
C
2059  IF(T.LT.CALCST) GO TO 207
      IF(ITURN.EQ.0.OR.IEND.EQ.1) GO TO 207
C
C----COMPUTE TURN ANGLE AND TIME
      XMI=XD1(2)/XD1(1)
      XMI1=(YWP(I+1)-YWP(I+2))/(XWP(I+1)-XWP(I+2))
      XINT=-(YWP(I+1)-X1(2)+XMI*X1(1)-XMI1*XWP(I+1))/(XMI1-XMI)
      YINT=XMI*XINT-XMI*X1(1)+X1(2)
      XINTNM=XINT/FEET
      YINTNM=YINT/FEET
      DTG=DSQRT((XINT-X1(1))**2+(YINT-X1(2))**2)
      TLEG1=DTG/VEG
      DELH=DABS(H(I+1)-HEA)
      IF(DELH.GT.180.) DELH=360.-DELH
      TO21=VAH*DELH*DEG/(2.0*G*TAN)
      IF(TO21/TAUH.GT.0.6) GO TO 2060
      TT21=TAUH*DSQRT(2.0*TO21/TAUH)
      GO TO 2061
C
C----COMPUTE TURN HALF PERIOD BY SOLVING EQUATION
C    USING THE NEWTON-RAPHSON METHOD
2060  TT21=TAUH*(1.0+TO21/TAUH)
2061  U=TT21/TAUH
      DO 2062 J=1,10
      F=U+DEXP(-U)-1.0-TO21/TAUH
      FP=1.0-DEXP(-U)
      UOLD=U
      U=UOLD-F/FP
      IF(DABS(TAUH*(U-UOLD)).LE.0.01) GO TO 2063
2062  CONTINUE
2063  TT21=TAUH*U
C
      DA1=VEG*(TT21-TO21)
      DC1=2.*VWH*TO21*DSIN(DEG*(THETA(I+1)-THETWH))
     1    *DSIN(DEG*(THETA(I+1)-HEG))
      DB1=VAH*VAH/G/TAN*(DSIN(DEG*(H(I+1)-HEA))*DSIN(DEG*(THETA(I+1)-
     2HEA))-(1.-DCOS(DEG*(H(I+1)-HEA)))*DCOS(DEG*(THETA(I+1)-HEA)))
     3/DABS(DSIN(DEG*(THETA(I+1)-HEA)))
```

```
      D61=DA1+DB1+DC1                                                   LENT2890
C                                                                       LENT2900
      TTURN1=D61/VEG                                                    LENT2910
      TSTPTN=2.0*(TT21-TAUH)                                            LENT2920
      TTOTTN=2.0*TT21                                                   LENT2930
      TG=TLEG1-TTURN1                                                   LENT2940
C                                                                       LENT2950
 207  IF(LNC.LE.LNCLIM) GO TO 208                                       LENT2960
      PRINT 1101                                                        LENT2970
      PRINT 1200                                                        LENT2980
      LNC=5                                                             LENT2990
 208  LNC=LNC+2                                                         LENT3000
      PRINT 1201,T,PHC,PH1,TLEG1,TTURN1,TO21,TT21,                      LENT3010
     1           HTG,HEG,HTA,HEA,                                       LENT3020
     2           ETRUE,EEST,PHD1,PHD2                                   LENT3030
C                                                                       LENT3040
C----T IS TIME                                                          LENT3050
C    TG IS TIME TO GO BEFORE BEGINNING COMMAND TURN                     LENT3060
C    ITH COUNTS TOTAL NUMBER OF SAMPLES ON THIS RUN                     LENT3070
      ITH=ITH+1                                                         LENT3080
      T=T+DT                                                            LENT3090
      IF(T.LT.CALCS1) GO TO 191                                         LENT3100
      IF(T.GE.TSTOP) GO TO 999                                          LENT3110
C                                                                       LENT3120
C----BRANCH TO 210 IF NOT EXECUTING OR COMPLETING A TURN                LENT3130
C    BRANCH TO 220 WHEN TIME TO BEGIN A TURN                            LENT3140
C    BRANCH TO 209 WHEN COMPLETING A TURN                               LENT3150
      IF(ITURN) 210,220,209                                             LENT3160
 209  IF(TG.LE.0.0.AND.IEND.NE.1) GO TO 219                             LENT3170
      GO TO 240                                                         LENT3180
 210  IF(TG.LE.0.0.AND.IEND.NE.1) GO TO 219                             LENT3190
C                                                                       LENT3200
C----GENERATE COURSE GUIDANCE TO CAUSE AIRCRAFT TO FLY                  LENT3210
C    TOWARD CURRENT LEG'S END POINT.  BANK ANGLE COMMAND IS             LENT3220
C    QUANTIZED TO NEAREST 15/128 DEGREE.  PHC=0 COMMAND SENT IF         LENT3230
C    HEADING ERROR TO CURRENT LEG END POINT IS LESS THAN HERMIN.        LENT3240
      IF(TG.GT.3.0*TAUH+DT.OR.IEND.EQ.1) GO TO 212                      LENT3250
      PHC=0.0                                                           LENT3260
      GO TO 191                                                         LENT3270
C                                                                       LENT3280
 212  HEGWPT=RAD*DATAN2((XWP(I+1)-X1(1)),(YWP(I+1)-X1(2)))              LENT3290
      HDEOLD=HDE                                                        LENT3300
      HDE=HEGWPT-HEG                                                    LENT3310
      IF(HDE.GT.180.) HDE=HDE-360.                                      LENT3320
      IF(HDE.LT.-180.) HDE=HDE+360.                                     LENT3330
      HDEDOT=(HDE-HDEOLD)/DT                                            LENT3340
      PHD1=G1*HDE                                                       LENT3350
      PHD2=G2*HDEDOT                                                    LENT3360
```

```
      PHD=PHD1+PHD2                                                     LENT3370
      IF(DABS(PHD).GT.PHILIM) PHD=PHILIM*PHD/DABS(PHD)                  LENT3380
C                                                                       LENT3390
      PHC=(PHD)*128.D0/15.D0+.5D0                                       LENT3400
      NHC=PHC                                                           LENT3410
      PHC=NHC                                                           LENT3420
      PHC=PHC*15.D0/128.D0                                              LENT3430
C                                                                       LENT3440
C                                                                       LENT3450
      IF(DABS(HDE).LT.HERMIN) PHC=0.0                                   LENT3460
      GO TO 191                                                         LENT3470
C                                                                       LENT3480
 219  ITURN=0                                                           LENT3490
C                                                                       LENT3500
C----NWLD IS SET GREATER THAN ZERO HERE TO SUPPRESS RADAR UPDATES       LENT3510
C      DURING A COMMAND TURN                                            LENT3520
      NWLD=1                                                            LENT3530
      TINTRN=0.0                                                        LENT3540
      IF(LNC.LE.LNCLIM) GO TO 2191                                      LENT3550
      PRINT 1101                                                        LENT3560
      LNC=0                                                             LENT3570
2191  LNC=LNC+3                                                         LENT3580
      PRINT 2001, T, TINTRN                                             LENT3590
 220  IF(TINTRN.GE.TSTPTN) GO TO 239                                    LENT3600
      TINTRN=TINTRN+DT                                                  LENT3610
      TTOTT1=TTOTTN                                                     LENT3620
C                                                                       LENT3630
C----COMPUTE DIRECTION OF COMMAND TURN                                  LENT3640
      DELH=HNEXT-HSTART                                                 LENT3650
      IF(DELH.GT.180.) DELH=DELH-360.                                   LENT3660
      IF(DELH.LT.-180.) DELH=DELH+360.                                  LENT3670
      IF(DELH.GT.0.) PHC=PHILIM                                         LENT3680
      IF(DELH.LT.0.) PHC=-PHILIM                                        LENT3690
      GO TO 191                                                         LENT3700
C                                                                       LENT3710
 239  ITURN=1                                                           LENT3720
      IF(LNC.LE.LNCLIM) GO TO 2391                                      LENT3730
      PRINT 1101                                                        LENT3740
      LNC=0                                                             LENT3750
2391  LNC=LNC+3                                                         LENT3760
      PRINT 2002, T,TINTRN                                              LENT3770
      PHC=0.                                                            LENT3780
C                                                                       LENT3790
C----BEGIN NEW LEG                                                      LENT3800
      I=I+1                                                             LENT3810
      HSTART=H(I)                                                       LENT3820
      HNEXT=H(I+1)                                                      LENT3830
      IF(I.EQ.NLEG) IEND=1                                              LENT3840
```

```
C
      IF(LNC.LE.LNCLIM-21) GO TO 2392
      PRINT 1101
      LNC=0
 2392 LNC=LNC+21
C
      PRINT 1100,I,XWPNM(I),YWPNM(I),XWPNM(I+1),YWPNM(I+1),PLENTH(I),
     1           THETA(I),VGH(I),H(I),RANGE(I),AZ(I),TLEG(I)
      PRINT 1200
C
  240 IF(TINTRN.GE.TTOTT1) GO TO 280
      TINTRN=TINTRN+DT
      GO TO 191
C
C----RESET TURN LOGIC FOR TURN AT END OF NEW LEG
  280 ITURN=-1
      NALD=-1
      IF(LNC.LE.LNCLIM) GO TO 285
      PRINT 1101
      LNC=0
  285 LNC=LNC+3
      PRINT 2003, T,TINTRN
      TINTRN=0.
      IF(IEND.EQ.1) TSTOP=T+TLEG(I)-2.0*TT21-2.0*TAUH
      GO TO 191
C
  999 CONTINUE
      RMSEST=DSQRT(SUMEST/FLOAT(IAVG))
      RMSETR=DSQRT(SUMETR/FLOAT(IAVG))
      PRINT 1260, RMSETR,RMSEST
C
C----EXECUTE PLOTS
      ITAB2=ITAB1+1
      ITAB3=ITAB1+2
      DXX=XMAX-XMIN
      DYY=YMAX-YMIN
      IF(DXX.GT.DYY) YMAX=YMIN+DXX
      IF(DXX.LT.DYY) XMAX=XMIN+DYY
      XXA(ITAB2)=XMAX
      XXA(ITAB3)=XMIN
      YYA(ITAB2)=YMAX
      YYA(ITAB3)=YMIN
      WRITE(6,1250) ATITLE
      CALL PLOTP(XXA,YYA,ITAB3,1)
      CALL PLOTP(XXB,YYB,ITAB1,2)
      CALL PLOTP(XXC,YYC,ITABA,3)
      PRINT 1101
      STOP
```

```
C
C----INPUT FORMATS
 100  FORMAT(I1,F9.2,7F10.4)
 110  FORMAT(2F10.3)
 111  FORMAT(4F10.3)
 112  FORMAT(6A8)
C
C----OUTPUT FORMATS
C
1000  FORMAT(1H1,48X,'AN/TPQ-27 SIMULATION',//,37X,'COARSE ',
     1'GUIDANCE MODE WITH KALMAN FILTERING',///,5X,'INITIAL CONDITIONS :
     2',//,8X,'TRUE WIND = ',F6.2,' FT/SEC AT ',F6.2,' DEGREES',
     3/,8X,'TRUE WIND COMPONENTS = ',2F6.2,/,
     48X,'ESTIMATED WIND = ',F6.2,' FT/SEC AT ',F6.2,' DEGREES',/,
     58X,'ESTIMATED WIND COMPONENTS = ',2F6.2,//,
     6/,8X,'RADAR DATA',//,10X,
     6'MEASUREMENT SIGMAS(R(FT.),AZ(DEG),EL(DEG)) = ',1P3D16.5,/,10X
     7,'MEASUREMENT BIASES(          "            ) = ' ,1P3D16.5,/,
     810X,'INITIAL VELOCITY MEASUREMENT VALUES = ',7X,1P3D16.5,/,
     910X,'RANDOM FORCING ASSUMPTION VALUES (SIGW) = ',3X,1P3D16.5,/,
     C10X,'SAMPLING INTERVAL FOR RADAR (DTRAD) = ',0PF6.3,///,
     18X,'AIRCRAFT DATA',//,
     210X,'TRUE INITIAL DISPLACEMENT FROM STARTING POINT (NM) = ',
     32F8.2,/,10X,'TRUE INITIAL GROUND VELOCITY = ',F8.2,
     4' FT/SEC AT ',F6.2,' DEGREES',/,
     510X,'TRUE INITIAL GROUND VELOCITY COMPONENTS = ',2F8.2,/,
     610X,'TRUE INITAL AIR VELOCITY = ',F8.2,' FT/SEC AT ',
     7F6.2,' DEGREES',/,10X,'ROLL RESPONSE PARAMETER (TB) = ',F5.3,///)
C
1001  FORMAT(8X,'TRACK DATA',//,10X,'NUMBER OF LEGS (NLEGS) = ',I2,/,
     110X,'POSITION OF FIRST LEG = ',F8.3,' NM AT ',F6.2,' DEGREES',/,
     210X,'COMPONENTS OF START POINT = ',2F8.3,//,15X,
     3///,8X,'CONTROL DATA',//,10X,'CONTROL INTERVAL = ',F6.3,/,
     410X,'MAXIMUM BANK ANGLE = ',F7.2,//,
     510X,'G1 = ',F8.3,/,10X,'G2 = ',F8.3,//,
     610X,'MIN HEADING ERROR ANGLE FOR COMMAND CORRECTION = ',
     7F8.3,///)
C
1100  FORMAT(//,4X,'LEG NUMBER',I2,//,10X,'LEG START POINT (X,Y) = ',
     12F10.4,/,10X,'LEG END POINT (X,Y) = ',2X,2F10.4,//,
     210X,'LEG LENGTH (NM) = ',2X,F10.2,/,10X,'LEG AZIMUTH (DEG) = ',
     3F10.2,//,10X,'DESIRED GROUND SPEED (FT/SEC) = ',F7.2,/,
     410X,'DESIRED AIR HEADING (DEG) = ',F6.2,//,
     510X,'AVG RANGE OF LEG FROM RADAR (NM) = ',3X,F10.2,/,
     610X,'AVG AZIMUTH OF LEG FROM RADAR (DEG) = ',F10.2,//,
     710X,'TLEG = ',F7.3,///)
C
1101  FORMAT(1H1)
```

```
C                                                                              LENT4810
1200  FORMAT(4X,'T',7X,'PHC',7X,'PH1',5X,'TLEG1',4X,'TTURN1',6X,'TO21'       LENT4820
     C,6X,'TT21',                                                              LENT4830
     14X,'HTG',4X,'HEG',4X,'HTA',4X,'HFA',4X,'ETRUE',5X,'EEST',5X,            LENT4840
     22X,'PHD1',5X,'PHD2',//)                                                  LENT4850
C                                                                              LENT4860
1201  FORMAT(1X,F6.2,6F10.4,4F7.2,4F9.4,/)                                    LENT4870
C                                                                              LENT4880
1250  FORMAT(1H1,10X,6A8,//,11X,6A8,///)                                       LENT4890
C                                                                              LENT4900
1260  FORMAT(///,10X,'RMS ETRUE = ',F10.6,//,11X,'RMS EEST = ',F10.6)          LENT4910
C                                                                              LENT4920
2001  FORMAT(10X,'STARTING TURN  T = ',F6.2,8X,'TINTRN = ',F6.2,//)            LENT4930
2002  FORMAT(10X,'TURN ENDING    T = ',F6.2,8X,'TINTRN = ',F6.2,//)            LENT4940
2003  FORMAT(10X,'TURN COMPLETE  T = ',F6.2,8X,'TINTRN = ',F6.2,//)            LENT4950
      END                                                                      LENT4960
C                                                                              LENT4970
C                                                                              LENT4980
C                                                                              LENT4990
C                                                                              LENT5000
C                                                                              LENT5010
C                                                                              LENT5020
C                                                                              LENT5030
C                                                                              LENT5040
      SUBROUTINE ARCRFT                                                        LENT5050
C                                                                              LENT5060
      IMPLICIT REAL*8 (A-H,O-Z)                                                LENT5070
C                                                                              LENT5080
      COMMON/AIRCOM/X3(3),XD3(3),WT(3),PHB,TB,DT,PHC,ITH                       LENT5090
C                                                                              LENT5100
      DATA DEG,G/57.29577951308232l,32.174049/                                 LENT5110
C                                                                              LENT5120
C----FIRST TIME THROUGH, ITH=0                                                 LENT5130
      IF(ITH)4,1,4                                                             LENT5140
C                                                                              LENT5150
C----SUBTRACT TRUE WIND VELOCITY                                               LENT5160
    1 SM1=XD3(1)-WT(1)                                                         LENT5170
      SM2=XD3(2)-WT(2)                                                         LENT5180
C                                                                              LENT5190
C----VT IS AIRSPEED OF A/C                                                     LENT5200
C    PHB IS AUTOPILOT BANK ANGLE BIAS                                          LENT5210
C    TB IS A/C RESPONSE LOOP TIME CONSTANT                                     LENT5220
C    PS IS INITIAL TURN ANGLE                                                  LENT5230
C    PSD IS INITIAL TURN RATE                                                  LENT5240
      VT=DSQRT(SM1*SM1+SM2*SM2)                                                LENT5250
      PHB=0.D0                                                                 LENT5260
      PH=PHB                                                                   LENT5270
      CA1=G/VT                                                                 LENT5280
```

```
      CA2=DEXP(-DT/TB)                                                  LENT5290
      CA3=CA1                                                           LENT5300
      CA4=DT*DT/(12.D0*DEG)                                             LENT5310
      CA5=TB*(1.D0-CA2)                                                 LENT5320
      DT3=.5D0*DT                                                       LENT5330
      PSD=CA1*PH                                                        LENT5340
      PS=DATAN2(SM1,SM2)                                                LENT5350
      CPS=DCOS(PS)                                                      LENT5360
      SPS=DSIN(PS)                                                      LENT5370
      PS=PS*DEG                                                         LENT5380
      RETURN                                                            LENT5390
C                                                                       LENT5400
C********************************************************************** LENT5410
C                                                                       LENT5420
C----ENTER MAIN CALCULATION STREAM                                      LENT5430
C                                                                       LENT5440
C----PHC IS NOW THE COMMAND BANK ANGLE AT T(N-1)                        LENT5450
C    PHN IS THE NEW BANK ANGLE                                          LENT5460
C    PSDN IS THE NEW TURNING RATE                                       LENT5470
C    PSN IS NEW HEADING ANGLE FOR VELOCITY WITH RESPECT TO WIND         LENT5480
    4 PSN=PHB+PHC                                                       LENT5490
      PHN=PSN+(PH-PSN)*CA2                                              LENT5500
      PSDN=CA1*PHN                                                      LENT5510
      PSN=PS+CA3*(PSN*DT+CA5*(PH-PSN))                                  LENT5520
      CPSN=PSN/DEG                                                      LENT5530
      SPSN=DSIN(CPSN)                                                   LENT5540
      CPSN=DCOS(CPSN)                                                   LENT5550
C                                                                       LENT5560
C----UPDATE VELOCITY VECTOR FOR NEW TURN ANGLE                          LENT5570
      XD3(1)=WT(1)+VT*SPSN                                              LENT5580
      XD3(2)=WT(2)+VT*CPSN                                              LENT5590
      XD3(3)=0.D0                                                       LENT5600
C                                                                       LENT5610
C----UPDATE POSITION VECTOR FOR NEW TURN ANGLE                          LENT5620
      SM1=DT3*(SPS+SPSN)-CA4*(CPSN*PSDN-CPS*PSD)                        LENT5630
      SM2=DT3*(CPS+CPSN)-CA4*(-SPSN*PSDN+SPS*PSD)                       LENT5640
      SM3=0.D0                                                          LENT5650
C                                                                       LENT5660
      X3(1)=DT*WT(1)+VT*SM1+X3(1)                                       LENT5670
      X3(2)=DT*WT(2)+VT*SM2+X3(2)                                       LENT5680
      X3(3)=X3(3)                                                       LENT5690
C----RESET VALUES FOR NEXT TIME IN SUBROUTINE                           LENT5700
      PH=PHN                                                            LENT5710
C                                                                       LENT5720
      PS=PSN                                                            LENT5730
      PSD=PSDN                                                          LENT5740
      SPS=SPSN                                                          LENT5750
      CPS=CPSN                                                          LENT5760
```

```
      RETURN
      END
C
C
C
C
C
C
C
C
      SUBROUTINE RADAR6
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*4 RAN(3)
C
      COMMON/AIRCOM/X3(3),XD3(3),WT(3),PHB,TB,DT,PHC,ITH
C
      COMMON/RADCOM/SIG(6),BIS(3),X1(3),XD1(3),PS1,PH1,WR(3),
     1               DTRAD,SIGW(3),IU,NWLD
C
      DIMENSION Q(6,6),GAMMA(6,6),ADUM(6,6),BDUM(6,6),W(6,6),
     1PP(6,6),PE(6,6),H(6,6),HT(6,6),R(6,6),PHI(6,6),SIG1(3),
     2PHITPN(6,6),G(6,6),XIDENT(6,6),X1P(3),XD1P(3),XDATA(3),DELX(3)
C
      EQUIVALENCE (XDATA(1),RR),(XDATA(2),A),(XDATA(3),E)
C
      DATA DEG,GG/57.295779513082321,32.174049/
      DATA PSDN,PSD1,PHN,PSN,SN1,SN2,SPSN,CPSN,VT1/9*0.D0/
      DATA ANGMIN,ANGMAX,EPS/0.0000001,1.5707963,0.0001/
C
      IF(ITH.NE.0) GO TO 50
C
      INT=DTRAD/DT+0.5
      DTCUM=DTRAD
C
C-----COMPUTE BANK ANGLE CONSTANTS FOR DETERMINISTIC CONTROL MOTION
      CAA2=DEXP(-DT/TB)
      CA4=DT**2/(12.D0*DEG)
      CA5=TB*(1.D0-CAA2)
      DT3=DT/2.D0
C
C-----GENERATE Q ARRAY
      DO 1 I=1,6
      DO 1 J=1,6
      PE(I,J)=0.D0
      BDUM(I,J)=0.D0
      ADUM(I,J)=0.D0
      PP(I,J)=0.D0
      G(I,J)=0.D0
```

```
      R(I,J)=0.D0                                                       LENT6250
      GAMMA(I,J)=0.D0                                                   LENT6260
      W(I,J)=0.D0                                                       LENT6270
    1 Q(I,J)=0.D0                                                       LENT6280
      DTRAD2=DTRAD**2/2.D0                                              LENT6290
      W(1,1)=SIGW(1)**2                                                 LENT6300
      W(2,2)=SIGW(2)**2                                                 LENT6310
      W(3,3)=SIGW(3)**2                                                 LENT6320
      GAMMA(1,1)=DTRAD2                                                 LENT6330
      GAMMA(2,1)=DTRAD                                                  LENT6340
      GAMMA(3,2)=DTRAD2                                                 LENT6350
      GAMMA(4,2)=DTRAD                                                  LENT6360
      GAMMA(5,3)=DTRAD2                                                 LENT6370
      GAMMA(6,3)=DTRAD                                                  LENT6380
      CALL TRANS(GAMMA,6,3,ADUM)                                        LENT6390
      CALL PROD(GAMMA,W,6,3,3,BDUM)                                     LENT6400
      CALL PROD(BDUM,ADUM,6,3,6,Q)                                      LENT6410
C                                                                       LENT6420
C----INITIALIZE COVARIANCE OF PREDICTION ARRAY (PP)                     LENT6430
      DO 2 I=1,6                                                        LENT6440
      DO 2 J=1,6                                                        LENT6450
      PP(I,J)=0.D0                                                      LENT6460
    2 IF(I.EQ.J) PP(I,J)=1.D06                                          LENT6470
C                                                                       LENT6480
C----COMPUTE MEASUREMENT MATRIX AND VARIANCE                            LENT6490
      DO 3 I=1,6                                                        LENT6500
      DO 3 J=1,6                                                        LENT6510
    3 H(I,J)=0.D0                                                       LENT6520
      H(1,1)=1.D0                                                       LENT6530
      H(2,3)=1.D0                                                       LENT6540
      H(3,5)=1.D0                                                       LENT6550
      CALL TRANS(H,3,6,HT)                                              LENT6560
C                                                                       LENT6570
      SIG1(1)=SIG(1)                                                    LENT6580
      SIG1(2)=SIG(2)/DEG                                                LENT6590
      SIG1(3)=SIG(3)/DEG                                                LENT6600
      VARR=SIG1(1)**2                                                   LENT6610
      VARP=SIG1(2)**2                                                   LENT6620
      VART=SIG1(3)**2                                                   LENT6630
C                                                                       LENT6640
C----COMPUTE STATE TRANSITION ARRAY (PHI)                               LENT6650
      DO 4 I=1,6                                                        LENT6660
      DO 4 J=1,6                                                        LENT6670
    4 PHI(I,J)=0.D0                                                     LENT6680
      PHI(1,1)=1.D0                                                     LENT6690
      PHI(1,2)=DTRAD                                                    LENT6700
      PHI(2,2)=1.D0                                                     LENT6710
      PHI(3,3)=1.D0                                                     LENT6720
```

```
      PHI(3,4)=DTRAD                                                    LENT6730
      PHI(4,4)=1.D0                                                     LENT6740
      PHI(5,5)=1.D0                                                     LENT6750
      PHI(5,6)=DTRAD                                                    LENT6760
      PHI(6,6)=1.D0                                                     LENT6770
      CALL TRANS(PHI,6,6,PHITRN)                                        LENT6780
C                                                                       LENT6790
C----CREATE IDENTITY ARRAY                                              LENT6800
      DO 8 I=1,6                                                        LENT6810
      DO 8 J=1,6                                                        LENT6820
      XIDENT(I,J)=0.D0                                                  LENT6830
    8 IF(I.EQ.J) XIDENT(I,J)=1.D0                                       LENT6840
C                                                                       LENT6850
C----BYPASS PRINT EQUATIONS FOR THIS PROGRAM                            LENT6860
      IF(ITH.GT.-10) GO TO 50                                           LENT6870
      PRINT 310,((H(I,J),J=1,6),I=1,6)                                  LENT6880
  310 FORMAT(1H1,///,10X,'H ARRAY',//,6(/,6F17.5))                      LENT6890
      PRINT 311,((GAMMA(I,J),J=1,6),I=1,6)                              LENT6900
  311 FORMAT(///,10X,'GAMMA ARRAY',//,6(/,6F17.5))                      LENT6910
      PRINT 312,((Q(I,J),J=1,6),I=1,6)                                  LENT6920
  312 FORMAT(///,10X,'Q ARRAY',//,6(/,6F17.5))                          LENT6930
      PRINT 313,((PP(I,J),J=1,6),I=1,6)                                 LENT6940
  313 FORMAT(///,10X,'PP ARRAY',//,6(/,6F17.5))                         LENT6950
      PRINT 314,((PE(I,J),J=1,6),I=1,6)                                 LENT6960
  314 FORMAT(///,10X,'PE ARRAY',//,6(/,6F17.5))                         LENT6970
      PRINT 316,((PHI(I,J),J=1,6),I=1,6)                                LENT6980
  316 FORMAT(///,10X,'PHI ARRAY',//,6(/,6F17.4))                        LENT6990
C********************************************************************** LENT7000
C                                                                       LENT7010
C----BEGIN NORMAL FILTER COMPUTATIONS                                   LENT7020
C                                                                       LENT7030
C----COMPUTE TRUE R,AZ,EL,ADD NOISE, AND COMPUTE NOISY MEASUREMENTS     LENT7040
C    IN CARTESIAN COORDINATES                                           LENT7050
   50 DTCUM=DTCUM+DT                                                    LENT7060
      IF(DTCUM.LT.(DTRAD-EPS)) GO TO 61                                 LENT7070
      IF(NWLD.GE.0.AND.ITH.NE.0) GO TO 61                               LENT7080
      RR=X3(1)*X3(1)+X3(2)*X3(2)                                        LENT7090
      A=DATAN2(X3(1),X3(2))                                             LENT7100
      E=DATAN2(X3(3),DSQRT(RR))                                         LENT7110
      RR=DSQRT(RR+X3(3)*X3(3))                                          LENT7120
      RM2=RR**2                                                         LENT7130
C                                                                       LENT7140
      CALL NORMAL(IU,RAN,3)                                             LENT7150
      DO 6 I=1,3                                                        LENT7160
    6 XDATA(I)=XDATA(I)+BIS(I)+SIG1(I)*RAN(I)                           LENT7170
C                                                                       LENT7180
      IF(DABS(A).LT.ANGMIN) A=ANGMIN                                    LENT7190
      IF(DABS(E).LT.ANGMIN) E=ANGMIN                                    LENT7200
```

```
      SIGNA=A/DABS(A)                                                   LENT7210
      SIGNE=E/DABS(E)                                                   LENT7220
      DELA=DABS(A-ANGMAX)                                               LENT7230
      DELE=DABS(E-ANGMAX)                                               LENT7240
      IF(DELA.LT.ANGMIN) A=SIGNA*ANGMAX                                 LENT7250
      IF(DELE.LT.ANGMIN) E=SIGNE*ANGMAX                                 LENT7260
C                                                                       LENT7270
      CA=DCOS(A)                                                        LENT7280
      SA=DSIN(A)                                                        LENT7290
      CE=DCOS(E)                                                        LENT7300
      SE=DSIN(E)                                                        LENT7310
      CA2=CA**2                                                         LENT7320
      SA2=SA**2                                                         LENT7330
      CE2=CE**2                                                         LENT7340
      SE2=SE**2                                                         LENT7350
C                                                                       LENT7360
      XDATA(3)=RR*SE                                                    LENT7370
      XDATA(2)=RR*CE*CA                                                 LENT7380
      XDATA(1)=RR*CE*SA                                                 LENT7390
C                                                                       LENT7400
      IF(ITH.NE.0) GO TO 63                                             LENT7410
      DO 62 I=1,3                                                       LENT7420
      X1P(I)=XDATA(I)                                                   LENT7430
      X1(I)=X1P(I)                                                      LENT7440
      XD1P(I)=SIG(I+3)                                                  LENT7450
   62 DELX(I)=0.D0                                                      LENT7460
      IF(NWLD.GE.0) GO TO 61                                            LENT7470
C                                                                       LENT7480
C----COMPUTE COVARIANCE OF MEASUREMENT ERROR ARRAY (R)                  LENT7490
   63 R(1,1)=RM2*(VART*SE2*SA2+VARP*CE2*CA2+VART*VARP*SE2*CA2)          LENT7500
     1       +VARR*CE2*SA2                                              LENT7510
      R(2,2)=RM2*(VART*SE2*CA2+VARP*CE2*CA2+VART*VARP*SE2*SA2)          LENT7520
     1       +VARR*CE2*CA2                                              LENT7530
      R(3,3)=RM2*VART*CE2+VARR*SE2                                      LENT7540
      R(1,2)=RM2*VART*(1.D0-VARP)*(SE2*SA*CA)                           LENT7550
     1        +(VARR-RM2*VARP)*(CE2*SA*CA)                              LENT7560
      R(2,1)=R(1,2)                                                     LENT7570
      R(1,3)=(VARR-RM2*VART)*SE*CE*SA                                   LENT7580
      R(3,1)=R(1,3)                                                     LENT7590
      R(2,3)=(VARR-RM2*VART)*SE*CE*CA                                   LENT7600
      R(3,2)=R(2,3)                                                     LENT7610
C                                                                       LENT7620
  121 IF(ITH.EQ.-10)  PRINT 317,((R(I,J),J=1,6),I=1,6)                  LENT7630
  317 FORMAT(///,10X,'R ARRAY ',//,6(/,6F17.5),///)                     LENT7640
C                                                                       LENT7650
C----COMPUTE GAIN MATRIX (G)                                            LENT7660
      DO 81 I=1,6                                                       LENT7670
      DO 81 J=1,6                                                       LENT7680
```

```
      BDUM(I,J)=0.DO                                                    LENT7690
   81 ADUM(I,J)=0.DO                                                    LENT7700
      CALL PROD(H,PP,3,6,6,ADUM)                                        LENT7710
      CALL PROD(ADUM,HT,3,6,3,BDUM)                                     LENT7720
      DO 83 I=1,6                                                       LENT7730
      DO 83 J=1,6                                                       LENT7740
   83 ADUM(I,J)=0.DO                                                    LENT7750
      CALL ADDSUB(BDUM,R,3,3,ADUM,1)                                    LENT7760
      DO 82 I=1,6                                                       LENT7770
      DO 82 J=1,6                                                       LENT7780
   82 BDUM(I,J)=0.DO                                                    LENT7790
      CALL INVERT(3,ADUM,BDUM,KER,6)                                    LENT7800
      IF(KER.EQ.2)  PRINT 308                                           LENT7810
  308 FORMAT(10X,25(1H*),'INVERSION SINGULARITY ENCOUNTERED')           LENT7820
      DO 84 I=1,6                                                       LENT7830
      DO 84 J=1,6                                                       LENT7840
   84 ADUM(I,J)=0.DO                                                    LENT7850
      CALL PROD(PP,HT,6,6,3,ADUM)                                       LENT7860
      CALL PROD(ADUM,BDUM,6,3,3,G)                                      LENT7870
      GNX=G(1,1)                                                        LENT7880
      GNY=G(3,2)                                                        LENT7890
      GNZ=G(5,3)                                                        LENT7900
C                                                                       LENT7910
C----COMPUTE COVARIANCE OF ESTIMATION ARRAY (PE)                        LENT7920
      DO 85 I=1,6                                                       LENT7930
      DO 85 J=1,6                                                       LENT7940
      ADUM(I,J)=0.DO                                                    LENT7950
   85 BDUM(I,J)=0.DO                                                    LENT7960
      CALL PROD(G,H,6,3,6,ADUM)                                         LENT7970
      CALL ADDSUB(XIDENT,ADUM,6,6,BDUM,-1)                              LENT7980
      CALL PROD(BDUM,PP,6,6,6,PE)                                       LENT7990
C                                                                       LENT8000
C----COMPUTE COVARIANCE OF PREDICTION ARRAY (PP)                        LENT8010
      DO 86 I=1,6                                                       LENT8020
      DO 86 J=1,6                                                       LENT8030
      ADUM(I,J)=0.DO                                                    LENT8040
   86 BDUM(I,J)=0.DO                                                    LENT8050
      CALL PROD(PHI,PE,6,6,6,ADUM)                                      LENT8060
      CALL PROD(ADUM,PHITRN,6,6,6,BDUM)                                 LENT8070
      CALL ADDSUB(BDUM,Q,6,6,PP,1)                                      LENT8080
C                                                                       LENT8090
C----BYPASS DETERMINISTIC FORCING EQUATIONS FOR FIRST                   LENT8100
C    SAMPLING PERIOD OF DTRAD                                           LENT8110
   61 IF(ITH.LE.INT) GO TO 66                                           LENT8120
C                                                                       LENT8130
C----COMPUTE NEW HEADING ANGLE AND HEADING ANGLE RATE, BASED ON         LENT8140
C    UNBIASED COMMANDS AS SENT FROM THE CONTROLLER.                     LENT8150
C    PHN IS THE NEW ROLL ANGLE                                          LENT8160
```

```
C     PSDN IS THE NEW TURNING ANGLE RATE
C     PSN IS THE NEW HEADING ANGLE
      PHN=PHC+(PH1-PHC)*CAA2
      CA1=GG/VT1
      PSDN=CA1*PHN
      DELPS1= CA1*(PHC*DT+CA5*(PH1-PHC))
      PS1DEX=DELPS1/DT
      PSN=PS1+DELPS1
C
      CPSN=PSN/DEG
      SPSN=DSIN(CPSN)
      CPSN=DCOS(CPSN)
C
C----COMPUTE STATE PREDICTION VECTOR
   66 XD1P(1)=VT1*SPSN
      XD1P(2)=VT1*CPSN
      XD1P(3)=XD1(3)
C
      DELX(1)=DT3*(XD1P(1)+SN1)-CA4*(XD1P(2)*PSDN-SN2*PSD1)
      DELX(2)=DT3*(XD1P(2)+SN2)-CA4*(-XD1P(1)*PSDN+SN1*PSD1)
      DELX(3)=DT*XD1(3)
C
      DO 110 I=1,3
      XD1P(I)=XD1P(I)+WR(I)
  110 X1P(I)=X1(I)+DT*WR(I)+DELX(I)
C
      IF(DTCUM.LT.DTRAD-EPS) GO TO 97
      DTCUM=0.D0
      IF(NWLD.LT.0) GO  TO 67
C
   97 DO 64 I=1,3
      X1(I)=X1P(I)
   64 XD1(I)=XD1P(I)
      GO TO 65
C
C----COMPUTE STATE ESTIMATION VECTOR
   67 E1=XDATA(1)-X1P(1)
      E2=XDATA(2)-X1P(2)
      E3=XDATA(3)-X1P(3)
C
      X1(1)=X1P(1)+G(1,1)*E1+G(1,2)*E2+G(1,3)*E3
      XD1(1)=XD1P(1)+G(2,1)*E1+G(2,2)*E2+G(2,3)*E3
      X1(2)=X1P(2)+G(3,1)*E1+G(3,2)*E2+G(3,3)*E3
      XD1(2)=XD1P(2)+G(4,1)*E1+G(4,2)*E2+G(4,3)*E3
      X1(3)=X1P(3)+G(5,1)*E1+G(5,2)*E2+G(5,3)*E3
      XD1(3)=XD1P(3)+G(6,1)*E1+G(6,2)*E2+G(6,3)*E3
C
C----COMPUTE VALUES FOR ENTERING DETERMINISTIC CONTROL
```

```
C     CALCULATIONS ON NEXT ITERATION                                    LENT8650
   65 IF(ITH.LT.INT) RETURN                                             LENT8660
      PH1=PHN                                                           LENT8670
      PSD1=PSDN                                                         LENT8680
      SN1=XD1(1)-WR(1)                                                  LENT8690
      SN2=XD1(2)-WR(2)                                                  LENT8700
      VT1=DSQRT(SN1*SN1+SN2*SN2)                                        LENT8710
      PS1=DATAN2(SN1,SN2)                                               LENT8720
      CPS=DCOS(PS1)                                                     LENT8730
      SPS=DSIN(PS1)                                                     LENT8740
      PS1=PS1*DEG                                                       LENT8750
      RETURN                                                            LENT8760
      END                                                               LENT8770
C                                                                       LENT8780
C                                                                       LENT8790
C                                                                       LENT8800
C                                                                       LENT8810
C                                                                       LENT8820
C                                                                       LENT8830
C                                                                       LENT8840
C                                                                       LENT8850
      SUBROUTINE MATCAL                                                 LENT8860
      REAL*8 A,AA,B,C,D,DD,S,X,Y                                        LENT8870
      DIMENSION AA(1),X(1),LL(6),MM(6),Y(6,6),S(1),D(1),                LENT8880
     1A(6,6),B(6,6),C(6,6)                                              LENT8890
C                                                                       LENT8900
C                                                                       LENT8910
      ENTRY INVERT(N,AA,X,KER,K)                                        LENT8920
C                                                                       LENT8930
C       THIS SUBROUTINE INVERTS THE MATRIX A AND LEAVES THE             LENT8940
C       RESULTS IN THE MATRIX X.    N AND K AREA THE ORDER OF THE MATRIX.LENT8950
C       IF KER EQUALS 2 THEN A SINGULARITY HAS BEEN DETECTED.           LENT8960
C                                                                       LENT8970
      DO 1 I=1,N                                                        LENT8980
      DO 1 J=1,N                                                        LENT8990
      IND=(I-1)*K+J                                                     LENT9000
    1 Y(I,J)=AA(IND)                                                    LENT9010
C     KER=1                                                             LENT9020
      N2=2*N                                                            LENT9030
      CALL ARRAY(2,N,N,6,6,Y,Y)                                         LENT9040
      CALL DMINV(Y,N,DD,LL,MM)                                          LENT9050
      CALL ARRAY(1,N,N,6,6,Y,Y)                                         LENT9060
      IF(DD.EQ.0) KER=2                                                 LENT9070
      DO 2 I=1,N                                                        LENT9080
      DO 2 J=1,N                                                        LENT9090
      IND=(I-1)*K+J                                                     LENT9100
    2 X(IND)=Y(I,J)                                                     LENT9110
      RETURN                                                            LENT9120
```

```
C
C
      ENTRY ADDSUB(A,B,N,M,C,ISIGN)
C
C        THIS SUBROUTINE ADDS(ISIGN=1) OR SUBTRACTS(ISIGN=-1)
C        THE NXM MATRICES A AND B (A+B OR A-B) ,
C        STORING THE RESULT IN C.
C
      DO 125 I=1,N
      DO 125 J=1,M
  125 C(I,J)=A(I,J)+FLOAT(ISIGN)*B(I,J)
      RETURN
C
C
      ENTRY PROD(A,B,N,M,L,C)
C
C        THIS SUBROUTINE COMPUTES THE MATRIX PRODUCT AB
C        AND STORES THE RESULT IN C.  A IS NXM,B IS MXL, AND C IS NXL.
C
      DO 3 I=1,N
      DO 3 J=1,L
    3 C(I,J)=0.D0
      DO 126 I=1,N
      DO 126 J=1,L
      DO 126 K=1,M
  126 C(I,J)=C(I,J)+A(I,K)*B(K,J)
      RETURN
C
C
      ENTRY TRANS(A,N,M,C)
C
C        THIS SUBROUTINE TRANSPOSES THE  NXM MATRIX A
C        AND STORES THE RESULT AS THE MXN MATRIX C.
C
      DO 127 I=1,N
      DO 127 J=1,M
  127 C(J,I)=A(I,J)
      RETURN
      END
      SUBROUTINE ARRAY(MODE,I,J,N,M,S,D)
      DIMENSION S(1),D(1)
      REAL*8 S,D
      NI=N-I
      IF(MODE-1) 100,100,120
  100 IJ=I*J+1
      NM=N*J+1
      DO 110 K=1,J
      NM=NM-NI
```

```
      DO 110 L=1,I                                                      LENT9610
      IJ=IJ-1                                                           LENT9620
      NM=NM-1                                                           LENT9630
110   D(NM)=S(IJ)                                                       LENT9640
      GO TO 140                                                         LENT9650
 120  IJ=0                                                              LENT9660
      NM=0                                                              LENT9670
      DO 130 K=1,J                                                      LENT9680
      DO 125 L=1,I                                                      LENT9690
      IJ=IJ+1                                                           LENT9700
      NM=NM+1                                                           LENT9710
 125  S(IJ)=D(NM)                                                       LENT9720
 130  NM=NM+MI                                                          LENT9730
 140  RETURN                                                            LENT9740
      END                                                               LENT9750
C                                                                       LENT9760
C                                                                       LENT9770
C                                                                       LENT9780
C                                                                       LENT9790
C                                                                       LENT9800
C                                                                       LENT9810
C                                                                       LENT9820
C                                                                       LENT9820
      BLOCK DATA                                                        LENT9840
      IMPLICIT REAL*8(Z)                                                LENT9850
      COMMON/AIRCOM/Z1(14)                                              LENT9860
      COMMON/RADCOM/Z2(26)                                              LENT9870
      DATA Z1/14*0.0/                                                   LENT9880
      DATA Z2/26*0.0/                                                   LENT9890
      END                                                               LENT9900
C                                                                       LENT9910
C                                                                       LENT9920
C                                                                       LENT9930
C                                                                       LENT9940
C                                                                       LENT9950
C                                                                       LENT9960
C                                                                       LENT9970
C                                                                       LENT9980
//GO.SYSIN DD *
3   50.      45.        500.       500.       50.0       50.0       45.0      45.0
 10.0        0.0
 10.0         90.
14.1414      225.
  15.0       2.0        2.0
 DESIRED,TRUE, AND ESTIMATED POSITION
```

LIST OF REFERENCES

1. Benedict, T. R. and Bordner, G. W., "Synthesis of an Optimal Set of Radar Track-While-Scan Smoothing Equations," IRE Transactions on Automatic Control, February 1962.

2. Mills, E. H., Applying the Kalman Filter to the Emitter Location Problem Using Airborne Angle of Arrival Information, MS Thesis, Naval Postgraduate School, Monterey, California, Marcy 1973.

3. Meditch, J. S., Stochastic Optimal Linear Estimation and Control, McGraw-Hill Book Co., Inc., 1969.

4. Demetry, J. S., Notes on the Theory and Application of Optimal Estimation, paper presented and copyrighted for use at the Naval Postgraduate School, Monterey, California, 1970.

5. Berkowitz, K., "AN/TPQ-27 Precision Guidance-Autopilot Bias Compensation," AN/TPQ-27 Program Memorandum SQM-3000-7, 5 March 1971.

6. "Design of AN/TPQ-27 Precision Guidance System," AN/TPQ-27 Program Memorandum SQM-3000-6, 24 July 1970.

7. Wylie, C. R. Jr., Advanced Engineering Mathematics, McGraw-Hill Book Co., Inc., 1966.

8. Hartnett, E. J., "Pre-Mission Flight Planning- I: Mathematical Analysis," AN/TPQ-27 Program Memorandum SQM-3000-17, 14 August 1973.

9. Shucker, S., "Coarse Guidance System Changes - I: Mathematics," AN/TPQ-27 Program Memorandum SQM-3000-18, 23 August 1973.

10. Aldrich, G. T. and Krabill, W. B., "An Application of Kalman Techniques to Aircraft and Missile Radar Tracking," Paper 72-838, AIAA Guidance and Control Conference, Stanford, California, August 1972.

11. Johnson, R. S., "SPECTRA 70 Program Specification," unnumbered AN/TPQ-27 Program Memorandum, 18 March 1970.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0212 Naval Postgraduate School Monterey, California 93940	2
3.	Department Chairman Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	1
4.	Professor H. A. Titus Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	20
5.	Professor S. Jaurequi Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	1
6.	LT Robert Eugene Lentz, USN SMC #1507 Naval Postgraduate School Monterey, California 93940	1

Thesis
L526 Lentz
c.1
156198
Improvement of AN/TPQ-27 filter and control techniques.